Fuel optimal manoeuvres for multiple spacecraft formation reconfiguration using multi-agent optimization

Guang Yang[1,†], Qingsong Yang[2,‡], Vikram Kapila[1,*,§], Daniel Palmer[3,¶] and Ravi Vaidyanathan[4,‖]

[1] *Department of Mechanical, Aerospace, and Manufacturing Engineering, Polytechnic University, Brooklyn, NY 11201, U.S.A.*
[2] *Department of Electrical Engineering and Computer Science, Case Western Reserve University, Cleveland, OH 44106, U.S.A.*
[3] *Department of Mathematics and Computer Science, John Carroll University, Cleveland, OH 44118, U.S.A.*
[4] *Orbital Research, Inc., 673 G Alpha Drive, Cleveland, OH 44143, U.S.A.*

SUMMARY

The Air Force Research Laboratory has identified multiple spacecraft formation flying as an enabling technology for several future space missions. A key benefit of formation flying is the ability to reconfigure the spacecraft formation to achieve different mission objectives. In this paper, generation of fuel optimal manoeuvres for spacecraft formation reconfiguration is modelled and analysed as a multi-agent optimal control problem. Multi-agent optimal control is quite different from the traditional optimal control for single agent. Specifically, in addition to fuel optimization for a single agent, multi-agent optimal control necessitates consideration of task assignment among agents for terminal targets in the optimization process. In this paper, we develop an efficient hybrid optimization algorithm to address such a problem. The proposed multi-agent optimal control methodology uses calculus of variation, task assignment, and parameter optimization at different stages of the optimization process. This optimization algorithm employs a distributed computational architecture. In addition, the task assignment algorithm, which guarantees the global optimal assignment of agents, is constructed using the celebrated principle of optimality from dynamic programming. A communication protocol is developed to facilitate decentralized

*Correspondence to: Vikram Kapila, Department of Mechanical, Aerospace and Manufacturing Engineering, Polytechnic University, Brooklyn, NY 11201, USA
[†] E-mail: gyang01@utopia.poly.edu
[‡] E-mail: qxy3@po.cwru.edu
[§] E-mail: vkapila@duke.poly.edu
[¶] E-mail: dpalmer@jcu.edu
[‖] E-mail: raviv@orbitalresearch.com

decision making among agents. Simulation results are included to illustrate the efficacy of the proposed multi-agent optimal control algorithm for fuel optimal spacecraft formation reconfiguration. Copyright © 2002 John Wiley & Sons, Ltd.

1. INTRODUCTION

Control of multiple, homogeneous/heterogeneous agents is an enabling technology for many military, aerospace, industrial, and commercial applications. For example, concepts of intelligent vehicle highway system [1] and free-flying aircraft [2, 3] rely on interacting ground and air vehicles, respectively. In addition, the problem of multiple mobile robots performing a single task jointly is dependent on coordination between the robots [4]. Reconnaissance and other military functions from space, air, ground, and sea may all greatly benefit from cooperation between spacecraft, aircraft, tanks, carriers, submarines, weapons, etc., [5–8]. Even as the paradigm of multi-agent missions holds enormous potential, significant issues concerning their coordinated control remain unresolved. For example, a number of interesting dynamics and control-related research problems arise in the control of multiple space vehicles such as mission/path planning, on-board autonomy, orbital-debris avoidance, distributed control architecture, communication and throughput constraints, etc.

A specific multiple aerospace vehicle application, which has recently received significant attention, is distributed spacecraft formation flying (DSFF). DSFF represents the concept of distributing the functionality and cost of a large, specialized spacecraft among multiple smaller, less-expensive, cooperative spacecraft. It has been identified as an enabling technology by the Air Force Research Laboratory and NASA for future space missions [9–15]. Specifically, as the Air Force positions itself to become an 'Air and Space Force', it is envisaged that the DSFF technology will facilitate critical elements of its vision, viz. virtual global awareness and rapid access to space [16]. For example, Air Force's TechSat-21 program [9, 10] relies on successful development and deployment of DSFF technologies. Similarly, NASA is seeking rapid advancements in DSFF technologies to enable the Earth Orbiter-I and the New Millennium Interferometer programs [6, 11–13], among others. In particular, the DSFF paradigm is being envisioned as a versatile, adaptable, and affordable space technology capable of accomplishing diverse, multiple missions such as synthetic aperture radars, enhanced stellar optical interferometers, virtual co-observing, stereo-imaging, correlated real-time sensing, and simultaneous multipoint probing [14, 17]. The DSFF architecture necessitates interacting spacecraft with system-wide common capabilities (communication, sensing, navigation, control, etc.), operating collectively to accomplish shared mission objectives [9, 10]. The implementation of the DSFF concept requires tight, autonomous, real-time control of the relative distance and attitude between the participating spacecraft.

From a decision theory and control design perspective, the DSFF problem can be decomposed into two different but closely related phases (i) *distributed spacecraft formation keeping and maintenance* (DSFKM) and (ii) *distributed spacecraft formation reconfiguration* (DSFR). The DSFKM control must be designed to enable the spacecraft to undergo periodic relative motion such that a relative spatial pattern persists for several orbital periods with minimal propellant expenditure [18]. For DSFKM, linear and nonlinear formation dynamic

models have been developed and a variety of low-level control designs have been proposed to guarantee the desired DSFKM performance [8, 15, 19–27]. Specifically, a linear formation dynamic model known as the Hill's equations is given in References [19, 20]. Hill's equations constitute the foundation for the application of various linear control techniques to the DSFKM problem [22, 23, 26, 27]. Recently, Sedwick *et al.* [15] identified the set of feasible initial conditions that annihilate the secular growth in time in the solution of Hill's equations; thus, yielding periodic relative motion for DSFKM. Based on the work in Reference [15], spatial patterns for formation design were proposed in Reference [24]. In Reference [8], a class of control laws was designed based on exact knowledge of the DSFF model that yields local asymptotic position tracking and global exponential attitude tracking. A persistent disturbance was employed in Reference [8] to account for model imperfection, measurement inaccuracies, etc. In addition, an adaptive position controller was developed in Reference [28] that compensates for unknown, constant disturbances while producing globally asymptotically decaying position tracking errors. More recently, [21, 29] proposed nonlinear controllers for a leader-follower DSFKM system, which ensure global asymptotic position tracking errors. Furthermore, a formation initialization scheme was developed in Reference [29], which in the ideal case yields a no-thrust, periodic relative motion between the leader–follower spacecraft pair, and serves as a desired, relative motion trajectory. Finally, in Reference [25], an analytical method was developed to identify families of formations invariant to the J_2 perturbation in the geopotential, a dominant cause for formation dispersion.

In contrast to the DSFKM problem, the DSFR problem has received scant attention (see References [8, 21, 27, 29, 30–32] for recent exceptions). However, as discussed above, DSFR is an essential component of the DSFF paradigm meriting utmost attention since it enables the spacecraft formation to adaptively assume a desired formation pattern as dictated by mission objectives. A critical issue in the DSFR problem is to determine fuel optimal manoeuvres when an initial spacecraft formation is directed to reconfigure to a new formation pattern. In the current literature [8, 21, 27, 29, 30, 32], the DSFR problem is addressed by utilizing low-level trajectory tracking controllers, which reconfigure a spacecraft formation by considering the initial formation to be an initial offset for the new formation pattern. The principal drawback of these control designs is their unpredictable and non-optimal fuel expenditure for different formation reconfiguration processes. In an alternative DSFR approach, fuel optimal manoeuvres for all the spacecraft can be generated on-board. Subsequently, low-level controllers can be used to track the fuel optimal manoeuvres in order to yield fuel optimal formation reconfiguration. In this framework, the fuel cost for each spacecraft manoeuvre can be calculated, on-demand, and the total fuel expenditure for formation reconfiguration is minimized. Recall that the linearized relative dynamics of spacecraft, given by Hill's equations, is known to approximate the dynamic behaviour of spacecraft relative motion fairly well for short-period manoeuvres and small spatial separation between spacecraft. It is anticipated that the typical spacecraft formation reconfiguration manoeuvres will be executed within few orbital periods with small spatial separation between spacecraft (few kilometers). Thus, in this paper, we utilize the linearized relative dynamics of spacecraft to construct an optimization algorithm that generates the fuel optimal formation reconfiguration trajectories with low computational effort.

The traditional theory of calculus of variation and optimal control can generate fuel optimal trajectories for a single spacecraft with various types of boundary conditions. However, for the DSFR problem we face the challenge of generating a set of fuel optimal manoeuvres for

multiple spacecraft simultaneously. Specifically, for the DSFR problem we must (i) determine an optimal manoeuvre time interval for each spacecraft to execute its fuel optimal manoeuvre and (ii) solve an assignment problem so that the position arrangement for the spacecraft in the new formation pattern is optimal. Finally, as shown in Section 2.5 of this paper, frequently, a given formation pattern may leave certain formation parameters unspecified, which must be selected to ensure fuel optimal spacecraft formation reconfiguration. Thus, the fuel optimal DSFR problem poses a multi-agent optimal control problem necessitating a hybrid optimization involving calculus of variation, task assignment, and parameter optimization. In this paper, we develop an optimization algorithm that solves such a problem efficiently. In addition, we propose a communication protocol to facilitate a distributed computational architecture for the optimization algorithm.

The contents of this paper are as follows. In Section 2, we present a mathematical model for spacecraft formation reconfiguration. The formation reconfiguration process is analysed qualitatively in Section 3. In Section 4, we develop an optimization algorithm to solve the fuel optimal spacecraft formation reconfiguration problem. Simulation results are included in Section 5 to illustrate the efficacy of the proposed optimization algorithm for spacecraft formation reconfiguration. Finally, concluding remarks are given in Section 6.

2. PROBLEM MODELLING AND ANALYSIS

2.1. Coordinate system description

In the current literature, the DSFF dynamics is typically characterized via a leader-fixed non-inertial coordinate frame [15, 21–24, 26, 27, 29, 31–33]. A schematic drawing of such a coordinate frame is given in Figure 1, where we make the following considerations: (i) an inertial coordinate frame $\{X, Y, Z\}$ is attached to the centre of the Earth; (ii) $R_\ell(t) \in \mathbb{R}^3$ denotes the position vector from the origin of the inertial coordinate frame to the leader spacecraft; (iii) in the leader-fixed non-inertial coordinate frame $\{x_\ell, y_\ell, z_\ell\}$, the y_ℓ-axis points along the direction of the vector $R_\ell(t)$, the z_ℓ-axis points along the direction of the orbital angular momentum of the leader spacecraft, and the x_ℓ-axis is placed such that $\{x_\ell, y_\ell, z_\ell\}$ forms a right-handed coordinate frame; and (iv) $\rho(t) \in \mathbb{R}^3$ denotes the position vector from the origin of the coordinate frame $\{x_l, y_l, z_l\}$ to the follower spacecraft.

Unfortunately, in a general problem setting, the leader-fixed non-inertial coordinate frame presents severe constraints. First, any acceleration of the leader spacecraft influences the basic dynamics of the formation significantly. In particular, since the origin of the $\{x_\ell, y_\ell, z_\ell\}$ coordinate frame is attached to the leader spacecraft and the $\{x_\ell, y_\ell, z_\ell\}$ coordinate frame rotates with the leader spacecraft's orbital motion, the follower spacecraft dynamics in such a frame is influenced by some non-inertial forces. If the leader spacecraft is in an exact Keplerian (circular or elliptical) orbital motion, the non-inertial forces entering the follower spacecraft dynamics can be expressed in a relatively simple form [19, 21, 22, 26, 27, 29, 32]. However, when the leader spacecraft undergoes a non-Keplerian motion, e.g. during an orbital manoeuvre, the non-inertial forces in the follower spacecraft dynamics become quite complicated so that one can hardly predict the follower spacecraft dynamics. Note that, in practice, the leader spacecraft's orbit may need to be changed from time to time as dictated by mission
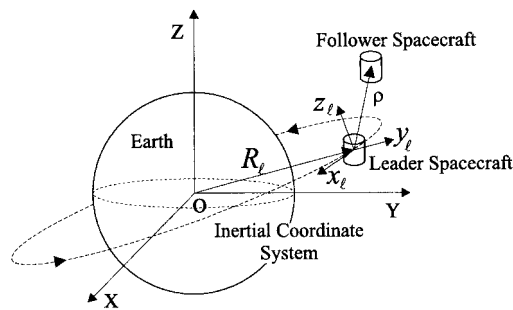
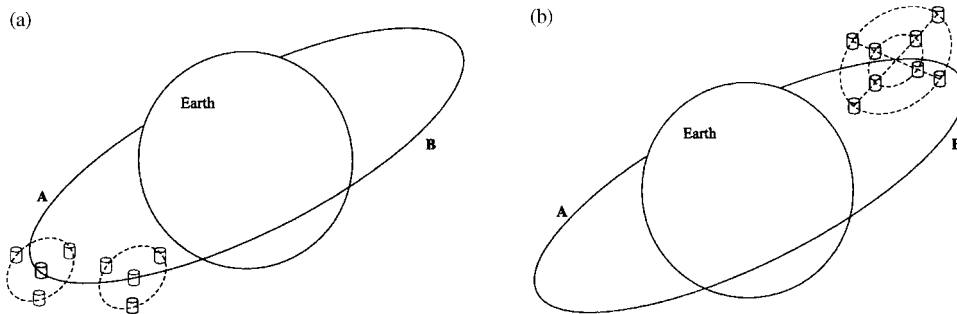Figure 1. Leader-fixed non-inertial coordinate frame in the DSFF system.

Figure 2. (a) Initial two cluster spacecraft formation; (b) reconfigured one cluster spacecraft formation.

requirements; thus, there is no reason to assume that the leader spacecraft will always be in a fixed Keplerian orbit.

Second, in many formation reconfiguration processes, a trivial choice for the leader spacecraft may not even exist. For example, consider two clusters of spacecraft coasting together in the beginning, as shown in Figure 2(a). Each cluster of spacecraft is composed of four spacecraft with one spacecraft located at the geometric centre and the other three spacecraft distributed on a thrust-free periodic relative orbit. The two spacecraft at the two geometric centres are in the same circular Earth orbit AB and their separation distance is large enough so that the two clusters do not overlap with one another; yet this distance is quite small compared to the radius of the orbit AB. Now, let all the spacecraft in the two clusters manoeuvre so that the two clusters combine into one. In the new formation, all the spacecraft are distributed on two concentric thrust-free periodic relative orbits (4 spacecraft on each) and the geometric centre of the new formation pattern (not occupied by any spacecraft) moves along the circular Earth orbit AB (as shown in Figure 2(b)). Clearly, in such a formation reconfiguration process there is no trivial choice for a leader spacecraft on which we can attach a non-inertial coordinate frame throughout the process.

In this paper, in order to overcome the aforementioned shortcomings of the leader-fixed non-inertial coordinate frame, we consider a non-inertial coordinate frame fixed on an imaginary leader spacecraft. Specifically, we attach a non-inertial coordinate frame $\{x_I, y_I, z_I\}$ to an imaginary point, the so called imaginary leader spacecraft, that is orbiting the Earth. Let $R_I(t) \in \mathbb{R}^3$ denote the position vector of the imaginary leader spacecraft in the Earth-centred inertial coordinate frame and let $\dot{R}_I(t) \in \mathbb{R}^3$ be the time derivative of $R_I(t)$. Then, we construct the $\{x_I, y_I, z_I\}$ coordinate frame such that the y_I-axis points along the direction of the vector $R_I(t)$, the z_I-axis points along the direction of the vector $R_I(t) \times \dot{R}_I(t)$, and the x_I-axis is placed such that $\{x_I, y_I, z_I\}$ forms a right-handed coordinate frame. In this paper, we let $R_I(t)$ follow an ideal Keplerian circular Earth orbit that represents the approximate average motion of the spacecraft formation. In such a coordinate frame, the formation dynamics is characterized analogous to the leader-fixed coordinate frame case, where a real leader spacecraft exists in an ideal Keplerian orbit. Furthermore, in the proposed $\{x_I, y_I, z_I\}$ coordinate frame, every spacecraft in the formation can manoeuvre freely without influencing the basic formation dynamics.

Now in the aforementioned formation reconfiguration example, we can choose an imaginary point, which is also in the circular Earth orbit AB and close enough to the initial formation, as an imaginary leader spacecraft. This imaginary leader spacecraft can serve as the origin of the coordinate frame $\{x_I, y_I, z_I\}$ throughout the formation reconfiguration process described above.

2.2. Thrust-free periodic trajectory for linearized spacecraft relative motion

In this paper, we limit ourselves to DSFF where an imaginary leader spacecraft can be chosen to follow an exact circular Earth orbit and all spacecraft in the formation fly along some thrust-free periodic trajectories relative to the $\{x_I, y_I, z_I\}$ coordinate frame. See References [15, 24, 27, 29, 30, 32] for further details on the design of thrust-free periodic formation trajectories. After the local linearization of the formation dynamics in the $\{x_I, y_I, z_I\}$ coordinate frame, the thrust-free periodic trajectories for spacecraft in the formation are the periodic solutions of the homogeneous linear Hill's equations written in the $\{x_I, y_I, z_I\}$ coordinate frame [15, 24, 27]. These trajectories form some closed periodic orbits in the $\{x_I, y_I, z_I\}$ coordinate frame [15, 24, 27]. Generally, in an n spacecraft formation, there are n' closed periodic orbits, where $n' \leqslant n$ since more than one spacecraft may be distributed on one closed periodic orbit. The distribution of the spacecraft on a particular closed periodic orbit can be specified by the phase difference of their periodic motion in this closed periodic orbit. This provides us a simple way to standardize the form of the parametric equations, which are used to describe formation patterns and spacecraft trajectories in the $\{x_I, y_I, z_I\}$ coordinate frame. Note that for a follower spacecraft the thrust-free periodic solution of the linear Hill's equation can be expressed as

$$x(t) = -2\frac{v_{y_0}}{\omega}\cos(\omega(t - t_0)) + X_C$$

$$y(t) = \frac{v_{y_0}}{\omega}\sin(\omega(t - t_0))$$

$$z(t) = z_0\cos(\omega(t - t_0)) + \frac{v_{z_0}}{\omega}\sin(\omega(t - t_0)) \tag{1}$$

where ω is the orbital angular velocity of the imaginary leader spacecraft, t_0 is the time when the follower spacecraft passes the x_I–z_I plane from the y_I^+ side to the y_I^- side, v_{y_0} and v_{z_0} are the y_I

and z_I velocity components, respectively, of the follower spacecraft relative to the $\{x_I, y_I, z_I\}$ coordinate frame at t_0, and z_0 is the z_I position component of the follower spacecraft relative to the $\{x_I, y_I, z_I\}$ coordinate frame at t_0. It is not difficult to see that the x_I velocity component of the follower spacecraft relative to the $\{x_I, y_I, z_I\}$ coordinate frame is zero at t_0. Since the follower spacecraft is passing the x_I–z_I plane from the y_I^+ side to the y_I^- side at t_0, v_{y_0} is negative. Note that this trajectory forms a closed elliptical orbit in the $\{x_I, y_I, z_I\}$ coordinate frame, with the geometric center located at $(X_C, 0, 0)$ [24, 33]. We call such an orbit a *primary formation orbit*. Let $\theta \triangleq -\omega t_0$ be the initial phase angle of the follower spacecraft in such a primary formation orbit, then its trajectory (1) can be expressed as

$$x(t) = -2\frac{v_{y_0}}{\omega}\cos(\omega t + \theta) + X_C$$

$$y(t) = \frac{v_{y_0}}{\omega}\sin(\omega t + \theta)$$

$$z(t) = z_0 \cos(\omega t + \theta) + \frac{v_{z_0}}{\omega}\sin(\omega t + \theta) \tag{2}$$

This form of parametric equations will be used as the standard form when we describe the thrust-free periodic spacecraft trajectories in the $\{x_I, y_I, z_I\}$ coordinate frame and will be mentioned as the *primary expression* in this paper. In the case where the relative motion of follower spacecraft is contained in the x_I–z_I plane, t_0 is arbitrary and $v_{y_0} = 0$.

Note that if another spacecraft is distributed on the same primary formation orbit, the primary expression for its thrust-free periodic trajectory will have the same y_I and z_I relative velocity components v_{y_0} and v_{z_0}, respectively, and the same z_I relative position component z_0, when it passes the x_I–z_I plane from the y_I^+ side to the y_I^- side. So, a primary formation orbit can be specified by a set of parameters $(X_C, v_{y_0}, v_{z_0}, z_0)$, which is called the *primary orbital parameter set* for the primary formation orbit.

2.3. Formation pattern

In this paper, an n spacecraft periodic spatial formation pattern is modelled as a set of n allowable positions that are distributed on n' primary formation orbits. We describe this set of allowable positions by primary expressions in the $\{x_I, y_I, z_I\}$ coordinate frame. Thus, this set of allowable positions represents the locations of the spacecraft in the formation as time-dependent functions. The number of allowable positions in the rth primary formation orbit, $r = 1, \ldots, n'$, is denoted as p_r, where $p_r \geqslant 1$ and $\sum_{r=1}^{n'} p_r = n$. Furthermore, the allowable positions distributed on the rth primary formation orbit, $r = 1, \ldots, n'$, are referenced as P_i^r, $i = 1, \ldots, p_r$.

Let the primary orbital parameter set for the rth primary formation orbit be $(X_{C_r}, v_{y_{0_r}}, v_{z_{0_r}}, z_{0_r})$. Then, the primary expression for the allowable position P_1^r is expressed as

$$x_{P_1^r}(t) = -2\frac{v_{y_{0_r}}}{\omega}\cos(\omega t + \theta_{P_1^r}) + X_{C_r}$$

$$y_{P_1^r}(t) = \frac{v_{y_{0_r}}}{\omega}\sin(\omega t + \theta_{P_1^r})$$

$$z_{P_1^r}(t) = z_{0_r}\cos(\omega t + \theta_{P_1^r}) + \frac{v_{z_{0_r}}}{\omega}\sin(\omega t + \theta_{P_1^r}) \tag{3}$$

In general, for the allowable position P_i^r, $i = 1, \ldots, p_r$, on the rth primary formation orbit, $r = 1, \ldots, n'$, the primary expression is given by

$$x_{P_i^r}(t) = -2 \frac{v_{y_{0_r}}}{\omega} \cos(\omega t + \theta_{P_1^r} + \beta_{i1_r}) + X_{C_r}$$

$$y_{P_i^r}(t) = \frac{v_{y_{0_r}}}{\omega} \sin(\omega t + \theta_{P_1^r} + \beta_{i1_r})$$

$$z_{P_i^r}(t) = z_{0_r} \cos(\omega t + \theta_{P_1^r} + \beta_{i1_r}) + \frac{v_{z_{0_r}}}{\omega} \sin(\omega t + \theta_{P_1^r} + \beta_{i1_r}) \tag{4}$$

where β_{i1_r}, $i = 1, \ldots, p_r$, is the phase difference between the periodic motions of the spacecraft located at the allowable position P_1^r and P_i^r. Note that $\beta_{11_r} = 0$ is immediate.

To illustrate the above concept, consider a 4 spacecraft, 1 primary orbit formation pattern whose projection on the ground is always a square, with each vertex being the projection of a spacecraft in the formation (see Figure 3). This formation requirement is satisfied by selecting the primary orbital parameter set for the primary formation orbit as $(X_{C_1}, u, 2u, 0)$, where $u < 0$. Then the formation pattern is a set of allowable positions $\{X_{P_1^1}(t), \ldots, X_{P_4^1}(t)\}$, where $X_{P_i^1}(t) \triangleq (x_{P_i^1}(t), y_{P_i^1}(t), z_{P_i^1}(t))$, $i = 1, \ldots, 4$. In addition, $x_{P_i^1}(t)$, $y_{P_i^1}(t)$, and $z_{P_i^1}(t)$ satisfy

$$x_{P_i^1}(t) = -2 \frac{u}{\omega} \cos(\omega t + \theta_{P_1^1} + \beta_{i1_1}) + X_{C_1}$$

$$y_{P_i^1}(t) = \frac{u}{\omega} \sin(\omega t + \theta_{P_1^1} + \beta_{i1_1})$$

$$z_{P_i^1}(t) = \frac{2u}{\omega} \sin(\omega t + \theta_{P_1^1} + \beta_{i1_1}) \tag{5}$$

where $i = 1, \ldots, 4$, $\beta_{11_1} = 0$, $\beta_{21_1} = \pi/2$, $\beta_{31_1} = \pi$, and $\beta_{41_1} = 3\pi/2$.

For notational convenience, in the balance of this paper, the n allowable positions in an n spacecraft formation pattern are indexed using only a subscript j, where $j \in \{1, \ldots, p_1, p_1 + 1, \ldots, p_1 + p_2, \ldots, p_1 + \cdots + p_{(n'-1)} + 1, \ldots, n\}$. Thus, an allowable position P_i^r in a subscript-only notation is referenced as P_j, which refers to the jth allowable position in the formation pattern, where $j = p_1 + \cdots + p_{(r-1)} + i$.

2.4. Spacecraft permutation in a formation pattern

In many cases, spacecraft locations may be interchangeable among all the allowable positions in a formation pattern. For example, in an n spacecraft formation, the spacecraft can be arranged onto the n allowable positions in $\mathscr{P}_n^n = n!$ many different ways, where $\mathscr{P}$ is the permutation operator defined as $\mathscr{P}_n^j \triangleq n!/(n-j)!$ [34] and '!' is the factorial operator [34]. Every such arrangement is called a *spacecraft permutation* for the formation pattern. Specifically, we label the spacecraft with n consecutive integers from 1 to n and use subscript S_i to refer to spacecraft i. Next, we define $X_{S_i}(t) \triangleq (x_{S_i}(t), y_{S_i}(t), z_{S_i}(t))$ to denote the thrust-free periodic trajectory for spacecraft i, $i = 1, \ldots, n$. Then a spacecraft permutation can be given by $H = [h_1, \ldots, h_n]$, where h_i, $i = 1, \ldots, n$, takes integer values from 1 to n, and $h_i \neq h_j$ for $i \neq j$. Specifically, for permutation $H = [h_1, \ldots, h_n]$, we have $X_{S_{h_1}} = X_{P_1}, \ldots, X_{S_{h_n}} = X_{P_n}$, i.e. spacecraft S_{h_1} is located at the 1st allowable position in the formation pattern, etc. For example, if we label the spacecraft with consecutive integers from 1 to 4 in the 4 spacecraft formation described before, we get $X_{S_1} = X_{P_1}$, $X_{S_2} = X_{P_2}$, $X_{S_3} = X_{P_3}$, and $X_{S_4} = X_{P_4}$, which yields a spacecraft permutation
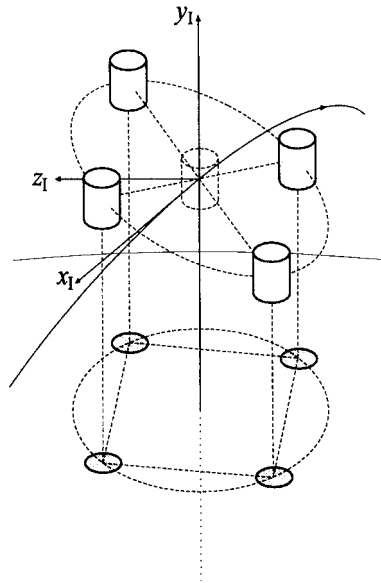
Figure 3. Spacecraft formation with square projection on the ground.

$H = [1, 2, 3, 4]$. However, if we let $X_{S_4} = X_{P_1}$, $X_{S_2} = X_{P_2}$, $X_{S_3} = X_{P_3}$, and $X_{S_1} = X_{P_4}$, we obtain a different spacecraft permutation $H = [4, 2, 3, 1]$ by interchanging the locations of spacecraft 1 and 4 while the formation pattern remains the same.

2.5. Formation reconfiguration

We describe a general formation reconfiguration as follows. Each spacecraft in an n spacecraft formation originally fly along their thrust-free periodic trajectories, which are described by some primary expressions in $\{x_I, y_I, z_I\}$. Such a formation will reconfigure to an m spacecraft formation whose pattern is described by a set of m allowable spacecraft positions, which are also expressed by primary expressions in $\{x_I, y_I, z_I\}$. For the new formation, m indicates the number of spacecraft that are needed to participate in the new formation pattern and it is restricted by $m \leqslant n$.

So far, we have assumed that the new formation pattern is uniquely specified. However, in many cases, the mission requirements for formation reconfiguration will not provide enough information to uniquely specify the new formation pattern, i.e. in the primary expressions for the new formation pattern some parameters may be chosen arbitrarily among some allowable values permitted by the mission requirements. Thus, there can be a family of new formation patterns that satisfy the mission requirements. For example, in the 4 spacecraft formation of Figure 3, if a square projection on the ground with side length $2\sqrt{2}u/\omega$ is the only requirement, then $\theta_{P_1^1}$ and X_{C_1} can be chosen freely in the primary expressions. In general, we assume that the mission requirements leave k independent parameters unspecified in the primary expressions for the new formation pattern. Let these k independent parameters be denoted by $\lambda_1, \ldots, \lambda_k$. In

addition, let $\Lambda_1, \ldots, \Lambda_k$ denote the sets of values for $\lambda_1, \ldots, \lambda_k$, respectively, which are permitted by the mission requirements. In this paper, we consider Λ_i, $i = 1, \ldots, k$, to be bounded sets. Let $\tilde{\lambda} \triangleq [\lambda_1, \ldots, \lambda_k]$, where $\tilde{\lambda} \in \tilde{\Lambda} \triangleq \Lambda_1 \times \cdots \times \Lambda_k$. Note that, together with the given mission requirements, $\tilde{\lambda} \in \tilde{\Lambda}$ uniquely specifies a new formation pattern.

For a spacecraft, a trajectory that smoothly connects the spacecraft's thrust-free periodic trajectory in the original formation and its thrust-free periodic trajectory in the new formation is referred to as a *spacecraft manoeuvre*. Specifically, a manoeuvre for spacecraft i to reach the jth allowable position in the new formation pattern that satisfies the given mission requirements and is also specified by $\tilde{\lambda}$, starting at the time instant t_{a_i} and ending at the time instant t_{b_i}, $t_{a_i} < t_{b_i}$, is defined by

$$\mathrm{m}_{i,j}(t; t_{a_i}, t_{b_i}, \tilde{\lambda}) \triangleq (x_{\mathrm{m}_{i,j}}(t; \tilde{\lambda}), y_{\mathrm{m}_{i,j}}(t; \tilde{\lambda}), z_{\mathrm{m}_{i,j}}(t; \tilde{\lambda})), \quad t_{a_i} \leqslant t \leqslant t_{b_i} \tag{6}$$

where $x_{\mathrm{m}_{i,j}}(t; \tilde{\lambda})$, $y_{\mathrm{m}_{i,j}}(t; \tilde{\lambda})$, and $z_{\mathrm{m}_{i,j}}(t; \tilde{\lambda})$ are the $x_\mathrm{I}, y_\mathrm{I}$, and z_I components, respectively, of the spacecraft manoeuvre, and are twice differentiable with respect to t since spacecraft's acceleration is finite. The statement that the manoeuvre starts at t_{a_i} and ends at t_{b_i} is mathematically interpreted to mean that the spacecraft position and velocity satisfy the boundary conditions

$$q_{\mathrm{m}_{i,j}}(t; \tilde{\lambda})\big|_{t=t_{a_i}} = q_{\mathrm{S}_i}(t)\big|_{t=t_{a_i}}, \quad \frac{\mathrm{d}}{\mathrm{d}t}(q_{\mathrm{m}_{i,j}}(t; \tilde{\lambda}))\big|_{t=t_{a_i}} = \frac{\mathrm{d}}{\mathrm{d}t}(q_{\mathrm{S}_i}(t))\big|_{t=t_{a_i}} \tag{7}$$

$$q_{\mathrm{m}_{i,j}}(t; \tilde{\lambda})\big|_{t=t_{b_i}} = q_{\mathrm{P}_j}(t; \tilde{\lambda})\big|_{t=t_{b_i}}, \quad \frac{\mathrm{d}}{\mathrm{d}t}(q_{\mathrm{m}_{i,j}}(t; \tilde{\lambda}))\big|_{t=t_{b_i}} = \frac{\mathrm{d}}{\mathrm{d}t}(q_{\mathrm{P}_j}(t; \tilde{\lambda}))\big|_{t=t_{b_i}} \tag{8}$$

where $q \in \{x, y, z\}$.

We emphasize that 'starting at t_{a_i}' does not mean that the manoeuvre $(x_{\mathrm{m}_{i,j}}(t; \tilde{\lambda}), y_{\mathrm{m}_{i,j}}(t; \tilde{\lambda}), z_{\mathrm{m}_{i,j}}(t; \tilde{\lambda}))$ is necessarily different from the initial trajectory $(x_{\mathrm{S}_i}(t), y_{\mathrm{S}_j}(t), z_{\mathrm{S}_i}(t))$ right after t_{a_i}. Similarly, 'ending at t_{b_i}' does not mean that the manoeuvre $(x_{\mathrm{m}_{i,j}}(t; \tilde{\lambda}), y_{\mathrm{m}_{i,j}}(t; \tilde{\lambda}), z_{\mathrm{m}_{i,j}}(t; \tilde{\lambda}))$ can not match the thrust-free trajectory $(x_{\mathrm{P}_j}(t; \tilde{\lambda}), y_{\mathrm{P}_j}(t; \tilde{\lambda}), z_{\mathrm{P}_j}(t; \tilde{\lambda}))$ in the new formation pattern before t_{b_i}. This enables us to extend the manoeuvre $\mathrm{m}_{i,j}(t; t_{a_i}, t_{b_i}, \tilde{\lambda})$ to a manoeuvre that transfers spacecraft i to the jth allowable position in the new formation pattern, starting at t'_{a_i} and ending at t'_{b_i}, where $t'_{a_i} \leqslant t_{a_i} < t_{b_i} \leqslant t'_{b_i}$. In particular, let

$$q_{\mathrm{m}_{i,j}^{\mathrm{e}}}(t; \tilde{\lambda}) \triangleq \begin{cases} q_{\mathrm{S}_i}(t) & t'_{a_i} \leqslant t \leqslant t_{a_i} \\ q_{\mathrm{m}_{i,j}}(t; \tilde{\lambda}) & t_{a_i} \leqslant t \leqslant t_{b_i}, \quad q \in \{x, y, z\} \\ q_{\mathrm{P}_j}(t; \tilde{\lambda}) & t_{b_i} \leqslant t \leqslant t'_{b_i} \end{cases} \tag{9}$$

then we define an extended manoeuvre as

$$\mathrm{m}_{i,j}^{\mathrm{e}}(t; t_{a_i}, t_{b_i}, t'_{a_i}, t'_{b_i}, \tilde{\lambda}) \triangleq (x_{\mathrm{m}_{i,j}^{\mathrm{e}}}(t; \tilde{\lambda}), y_{\mathrm{m}_{i,j}^{\mathrm{e}}}(t; \tilde{\lambda}), z_{\mathrm{m}_{i,j}^{\mathrm{e}}}(t; \tilde{\lambda})), t'_{a_i} \leqslant t \leqslant t'_{b_i}, t'_{a_i} \leqslant t_{a_i}, t_{b_i} \leqslant t'_{b_i} \tag{10}$$

which 'starts at t'_{a_i}' and 'ends at t'_{b_i}'. Note that the manoeuvre 'starts at t'_{a_i}' and 'ends at t'_{b_i}' follows from the fact that the boundary conditions (7) and (8) are automatically satisfied at t'_{a_i} and t'_{b_i}, respectively, by the way we construct $(x_{\mathrm{m}_{i,j}^{\mathrm{e}}}(t; \tilde{\lambda}), y_{\mathrm{m}_{i,j}^{\mathrm{e}}}(t; \tilde{\lambda}), z_{\mathrm{m}_{i,j}^{\mathrm{e}}}(t; \tilde{\lambda}))$ in (9). We call $\mathrm{m}_{i,j}^{\mathrm{e}}(t; t_{a_i}, t_{b_i}, t'_{a_i}, t'_{b_i}, \tilde{\lambda})$ a *primary extension* of the manoeuvre $\mathrm{m}_{i,j}(t; t_{a_i}, t_{b_i}, \tilde{\lambda})$ from the time interval $[t_{a_i}, t_{b_i}]$ to the time interval $[t'_{a_i}, t'_{b_i}]$. Figure 4 illustrates the notion of extended manoeuvre.
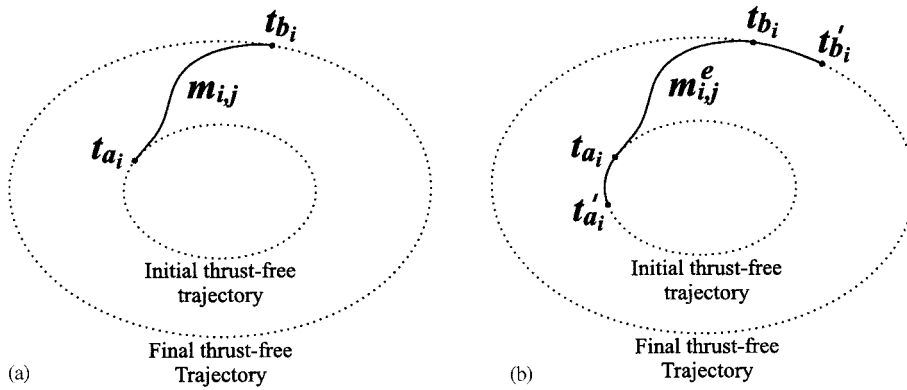

Figure 4. (a) Original manoeuvre; (b) extended manoeuvre.

3. FUEL OPTIMAL SPACECRAFT FORMATION RECONFIGURATION: MATHEMATICAL PRELIMINARIES

In this section, we begin by considering the fuel cost for spacecraft manoeuvres and their primary extensions to lay the foundation for developing the fuel optimal formation reconfiguration manoeuvres. First, let the fuel cost for spacecraft i to execute manoeuvre $m_{i,j}(t; t_{a_i}, t_{b_i}, \tilde{\lambda})$ be denoted as $I(m_{i,j}(t; t_{a_i}, t_{b_i}, \tilde{\lambda}))$. The specific form of this fuel cost is related to the formation dynamics model and the spacecraft thrust model as discussed in Section 4.2. Next, note that for the extended manoeuvre $(x_{m_{i,j}^e}(t; \tilde{\lambda}), y_{m_{i,j}^e}(t; \tilde{\lambda}), z_{m_{i,j}^e}(t; \tilde{\lambda}))$ in (9), the trajectory extensions lie on the thrust-free periodic trajectories of spacecraft i in the original formation and in the new formation. Thus, we have the following useful property for the fuel cost of the spacecraft manoeuvre $m_{i,j}(t; t_{a_i}, t_{b_i}, \tilde{\lambda})$ and its primary extension $m_{i,j}^e(t; t_{a_i}, t_{b_i}, t'_{a_i}, t'_{b_i}, \tilde{\lambda})$, even without the explicit knowledge of its specific form,

$$I(m_{i,j}^e(t; t_{a_i}, t_{b_i}, t'_{a_i}, t'_{b_i}, \tilde{\lambda})) = I(m_{i,j}(t; t_{a_i}, t_{b_i}, \tilde{\lambda})), \quad t'_{a_i} \leqslant t_{a_i} < t_{b_i} \leqslant t'_{b_i} \tag{11}$$

Next, we assume that each spacecraft in the formation is functionally identical, i.e. every spacecraft has the capability to perform the same tasks if required. Then, two different spacecraft permutations for the new formation pattern will not affect the formation performance. Therefore, in the n to m formation reconfiguration process, a spacecraft starting at its initial thrust-free periodic trajectory can manoeuvre to different allowable positions in the new formation pattern for different spacecraft permutations.

Before proceeding, for the time interval $[t_a, t_b]$, we define

$$T_m \triangleq \{[t_{a_{h_1}}, t_{b_{h_1}}], \dots, [t_{a_{h_m}}, t_{b_{h_m}}]\}, \quad t_a \leqslant t_{a_{h_j}} < t_{b_{h_j}} \leqslant t_b, \quad j = 1, \dots, m \tag{12}$$

$$\bar{T}_m \triangleq \{[t_{a_{h_1}}, t_{b_{h_1}}], \dots, [t_{a_{h_m}}, t_{b_{h_m}}]\}, \quad t_{a_{h_j}} = t_a, t_{b_{h_j}} = t_b, \quad j = 1, \dots, m \tag{13}$$

manoeuvre set $\mathrm{M}(t; H, T_m, \tilde{\lambda})$ yields

$$
\begin{aligned}
I_T(\mathrm{M}^e(t; H, T_m, \bar{T}'_m, \tilde{\lambda})) &= \sum_{j=1}^{m} I(\mathrm{m}^e_{h_j,j}(t; t_{a_{h_j}}, t_{b_{h_j}}, t'_a, t'_b, \tilde{\lambda})) \\
&= \sum_{j=1}^{m} I(\mathrm{m}_{h_j,j}(t; t_{a_{h_j}}, t_{b_{h_j}}, \tilde{\lambda})) \\
&= I_T(\mathrm{M}(t; H, T_m, \tilde{\lambda}))
\end{aligned}
\tag{21}
$$

i.e. the total fuel cost for a general manoeuvre set $\mathrm{M}(t; H, T_m, \tilde{\lambda})$ is the same as the total fuel cost for its time-regulated primary extension $\mathrm{M}^e(t; H, T_m, \bar{T}'_m, \tilde{\lambda})$ for any $t'_a \leqslant t_a \leqslant \min_{j=1,\dots,m} t_{a_{h_j}}$ and $t'_b \geqslant t_b \geqslant \max_{j=1,\dots,m} t_{b_{h_j}}$.

Among all the manoeuvres $\mathrm{m}_{i,j}(t; t_{a_i}, t_{b_i}, \tilde{\lambda})$ that start at t_{a_i} and end at t_{b_i}, $t_{a_i} < t_{b_i}$, and transfer spacecraft i to the jth allowable position in the new formation pattern, which is specified by $\tilde{\lambda}$, we define a *fuel optimal manoeuvre* $\mathrm{m}^*_{i,j}(t; t_{a_i}, t_{b_i}, \tilde{\lambda})$ as a manoeuvre whose fuel cost is smaller than (or at worst equal to) the fuel cost for any other manoeuvre, i.e.

$$
I(\mathrm{m}^*_{i,j}(t; t_{a_i}, t_{b_i}, \tilde{\lambda})) \leqslant I(\mathrm{m}_{i,j}(t; t_{a_i}, t_{b_i}, \tilde{\lambda}))
\tag{22}
$$

for all $\mathrm{m}_{i,j}(t; t_{a_i}, t_{b_i}, \tilde{\lambda})$.

Next, for a specified spacecraft permutation H, a set of chosen independent parameters $\tilde{\lambda}$ for the new formation pattern, and a set of manoeuvre time intervals T_m for all spacecraft in a general manoeuvre set, we define a *fuel optimal manoeuvre set* as

$$
\mathrm{M}^*(t; H, T_m, \tilde{\lambda}) \triangleq \{\mathrm{m}^*_{h_1,1}(t; t_{a_{h_1}}, t_{b_{h_1}}, \tilde{\lambda}), \dots, \mathrm{m}^*_{h_m,m}(t; t_{a_{h_m}}, t_{b_{h_m}}, \tilde{\lambda})\}
\tag{23}
$$

Since the total fuel cost for a manoeuvre set is the sum of the fuel cost for each spacecraft manoeuvre in that manoeuvre set, using (22), we obtain

$$
\begin{aligned}
I_T(\mathrm{M}^*(t; H, T_m, \tilde{\lambda})) &= \sum_{j=1}^{m} I(\mathrm{m}^*_{h_j,j}(t; t_{a_{h_j}}, t_{b_{h_j}}, \tilde{\lambda})) \\
&\leqslant \sum_{j=1}^{m} I(\mathrm{m}_{h_j,j}(t; t_{a_{h_j}}, t_{b_{h_j}}, \tilde{\lambda})) \\
&= I_T(\mathrm{M}(t; H, T_m, \tilde{\lambda}))
\end{aligned}
\tag{24}
$$

for all $\mathrm{M}(t; H, T_m, \tilde{\lambda})$. Thus, it follows from (24) that the total fuel cost for the fuel optimal manoeuvre set $\mathrm{M}^*(t; H, T_m, \tilde{\lambda})$ is smaller than (or at worst equal to) the total fuel cost for any other manoeuvre set $\mathrm{M}(t; H, T_m, \tilde{\lambda})$, which transfers m spacecraft from their initial thrust-free trajectories in the original formation to the m allowable positions according to the spacecraft permutation H in the new formation pattern specified by $\tilde{\lambda}$ and whose manoeuvre time intervals are given by T_m.

Note that the total fuel cost $I_T(\mathrm{M}^*(t; H, T_m, \tilde{\lambda}))$ is a function of H, T_m, and $\tilde{\lambda}$. In order to minimize the total fuel cost for a spacecraft formation reconfiguration process, we must find optimal H^*, T_m^*, and $\tilde{\lambda}^*$, such that the total fuel cost for $\mathrm{M}^*(t; H^*, T_m^*, \tilde{\lambda}^*)$ is smaller than (or at worst equal to) the total fuel cost for $\mathrm{M}^*(t; H, T_m, \tilde{\lambda})$ with any other choice of H, T_m, and $\tilde{\lambda}$; i.e.

$$
I_T(\mathrm{M}^*(t; H^*, T_m^*, \tilde{\lambda}^*)) \leqslant I_T(\mathrm{M}^*(t; H, T_m, \tilde{\lambda}))
\tag{25}
$$

Next, for a specified T_m, a choice of H and $\tilde{\lambda}$ is called *conditionally optimal* if the total fuel cost for $M^*(t; H^*_{T_m}, T_m, \tilde{\lambda}^*_{T_m})$ is smaller than (or at worst equal to) the total fuel cost for $M^*(t; H, T_m, \tilde{\lambda})$ with any other choice of H and $\tilde{\lambda}$, with the specified T_m, i.e.

$$I_T(M^*(t; H^*_{T_m}, T_m, \tilde{\lambda}^*_{T_m})) \leqslant I_T(M^*(t; H, T_m, \tilde{\lambda})) \tag{26}$$

Now we present the principal result of this section.

Theorem 3.1
On the time interval $[t_a, t_b]$, for arbitrary T_m,

$$I_T(M^*(t; H^*_{\bar{T}_m}, \bar{T}_m, \tilde{\lambda}^*_{\bar{T}_m})) \leqslant I_T(M^*(t; H^*_{T_m}, T_m, \tilde{\lambda}^*_{T_m})) \tag{27}$$

where T_m and $\bar{T}_m$ are given according to (12) and (13), respectively.

Proof
First, note that $H^*_{\bar{T}_m}$ and $\tilde{\lambda}^*_{\bar{T}_m}$ in (27) represent the conditionally optimal choice of H and $\tilde{\lambda}$ for the specified $\bar{T}_m$. Thus, according to (26), we obtain,

$$I_T(M^*(t; H^*_{\bar{T}_m}, \bar{T}_m, \tilde{\lambda}^*_{\bar{T}_m})) \leqslant I_T(M^*(t; H, \bar{T}_m, \tilde{\lambda})) \tag{28}$$

for all H and $\tilde{\lambda}$. Next, assume that (27) does not hold for some $\hat{T}_m$, i.e. there exists a $\hat{T}_m = \{[\hat{t}_{a_{h_1}}, \hat{t}_{b_{h_1}}], \ldots, [\hat{t}_{a_{h_m}}, \hat{t}_{b_{h_m}}]\}$, $t_a \leqslant \hat{t}_{a_{h_j}} < \hat{t}_{b_{h_j}} \leqslant t_b, j = 1, \ldots, m$, such that

$$I_T(M^*(t; H^*_{\bar{T}_m}, \bar{T}_m, \tilde{\lambda}^*_{\bar{T}_m})) > I_T(M^*(t; H^*_{\hat{T}_m}, \hat{T}_m, \tilde{\lambda}^*_{\hat{T}_m})) \tag{29}$$

Now for the manoeuvre set $M^*(t; H^*_{\hat{T}_m}, \hat{T}_m, \tilde{\lambda}^*_{\hat{T}_m})$ and its time-regulated primary extension $M^{*e}(t; H^*_{\hat{T}_m}, \hat{T}_m, \bar{T}_m, \tilde{\lambda}^*_{\hat{T}_m})$, (21) yields

$$I_T(M^{*e}(t; H^*_{\hat{T}_m}, \hat{T}_m, \bar{T}_m, \tilde{\lambda}^*_{\hat{T}_m})) = I_T(M^*(t; H^*_{\hat{T}_m}, \hat{T}_m, \tilde{\lambda}^*_{\hat{T}_m})) \tag{30}$$

Next, we define a time-regulated manoeuvre set $M(t; H^*_{\hat{T}_m}, \bar{T}_m, \tilde{\lambda}^*_{\hat{T}_m}) \triangleq M^{*e}(t; H^*_{\hat{T}_m}, \hat{T}_m, \bar{T}_m, \tilde{\lambda}^*_{\hat{T}_m})$. It now follows that

$$I_T(M(t; H^*_{\hat{T}_m}, \bar{T}_m, \tilde{\lambda}^*_{\hat{T}_m})) = I_T(M^{*e}(t; H^*_{\hat{T}_m}, \hat{T}_m, \bar{T}_m, \tilde{\lambda}^*_{\hat{T}_m})) \tag{31}$$

Combining (29)–(31), we obtain

$$I_T(M^*(t; H^*_{\bar{T}_m}, \bar{T}_m, \tilde{\lambda}^*_{\bar{T}_m})) > I_T(M(t; H^*_{\hat{T}_m}, \bar{T}_m, \tilde{\lambda}^*_{\hat{T}_m})) \tag{32}$$

It follows from (24) that

$$I_T(M(t; H^*_{\hat{T}_m}, \bar{T}_m, \tilde{\lambda}^*_{\hat{T}_m})) \geqslant I_T(M^*(t; H^*_{\hat{T}_m}, \bar{T}_m, \tilde{\lambda}^*_{\hat{T}_m})) \tag{33}$$

Therefore, there exist $H^*_{\hat{T}_m}$ and $\tilde{\lambda}^*_{\hat{T}_m}$ such that

$$I_T(M^*(t; H^*_{\bar{T}_m}, \bar{T}_m, \tilde{\lambda}^*_{\bar{T}_m})) > I_T(M^*(t; H^*_{\hat{T}_m}, \bar{T}_m, \tilde{\lambda}^*_{\hat{T}_m})) \tag{34}$$

which contradicts (28). Consequently, (27) holds for all T_m. □

According to Theorem 3.1, since (27) holds for arbitrary T_m, it holds for T^*_m, which is an optimal choice of T_m. Thus

$$I_T(M^*(t; H^*_{\bar{T}_m}, \bar{T}_m, \tilde{\lambda}^*_{\bar{T}_m})) \leqslant I_T(M^*(t; H^*_{T^*_m}, T^*_m, \tilde{\lambda}^*_{T^*_m})) \tag{35}$$

Since $H^*_{T^*_m}$ and $\tilde{\lambda}^*_{T^*_m}$ are conditionally optimal for T^*_m, which is the optimal choice of T_m, it follows that

$$I_T(\mathbf{M}^*(t; H^*_{T^*_m}, T^*_m, \tilde{\lambda}^*_{T^*_m})) \leqslant I_T(\mathbf{M}^*(t; H^*_{T_m}, T_m, \tilde{\lambda}^*_{T_m}))$$

$$\leqslant I_T(\mathbf{M}^*(t; H, T_m, \tilde{\lambda})) \tag{36}$$

for all H, T_m, and $\tilde{\lambda}$. Combining (35) and (36), we obtain

$$I_T(\mathbf{M}^*(t; H^*_{\bar{T}_m}, \bar{T}_m, \tilde{\lambda}^*_{\bar{T}_m})) \leqslant I_T(\mathbf{M}^*(t; H, T_m, \tilde{\lambda})) \tag{37}$$

for all H, T_m, and $\tilde{\lambda}$.

Corollary 3.1
On the time interval $[t_a, t_b]$, $H^*_{\bar{T}_m}$, $\bar{T}_m$, and $\tilde{\lambda}^*_{\bar{T}_m}$ are optimal in the sense that the total fuel cost for $\mathbf{M}^*(t; H^*_{\bar{T}_m}, \bar{T}_m, \tilde{\lambda}^*_{\bar{T}_m})$ is smaller than (or at worst equal to) the total fuel cost for $\mathbf{M}^*(t; H, T_m, \tilde{\lambda})$ with any other choice of H, T_m, and $\tilde{\lambda}$.

Now we consider the practical implication of this result to a spacecraft formation reconfiguration process, where, given by the mission requirement, the earliest allowable starting time and the latest allowable ending time for spacecraft manoeuvres are t_a and t_b, respectively. The significance of Corollary 3.1 is that our search space for the optimal choice of H, T_m, and $\tilde{\lambda}$ can be reduced to a considerably smaller search space for the conditionally optimal choice of H and $\tilde{\lambda}$ with the specified $\bar{T}_m$ when we minimize the total fuel cost $I_T(\mathbf{M}^*(t; H, T_m, \tilde{\lambda}))$ for a spacecraft formation reconfiguration process. In particular, according to Corollary 3.1, on the time interval $[t_a, t_b]$, in order to minimize the total fuel cost, we do not have to consider all general manoeuvre sets whose spacecraft manoeuvres may start and end at arbitrary time instants between t_a and t_b. Instead, it suffices to let every spacecraft manoeuvre start at t_a and end at t_b and obtain the fuel optimal manoeuvre set with the conditionally optimal choice of H and $\tilde{\lambda}$.

4. FUEL OPTIMAL SPACECRAFT FORMATION RECONFIGURATION: MULTI-AGENT OPTIMIZATION ALGORITHM ARCHITECTURE

In Section 3, we analysed the fuel optimal spacecraft formation reconfiguration problem qualitatively. In this section, we develop an architecture for the optimization algorithm that generates the fuel optimal manoeuvre set for spacecraft formation reconfiguration.

4.1. Multi-agent optimization algorithm architecture overview

Given the set $\bar{T}_m$, to minimize the total fuel cost for formation reconfiguration, our objective is to find a conditionally optimal choice of H and $\tilde{\lambda}$ and the corresponding fuel optimal manoeuvre set. This conditional optimization problem is decomposed into a two step process. In the first step, for any given choice of $\tilde{\lambda} \in \tilde{\Lambda}$, we find a conditionally optimal $H^*_{\tilde{\lambda}}$ such that

$$I_T(\mathbf{M}^*(t; H^*_{\tilde{\lambda}}, \bar{T}_m, \tilde{\lambda})) \leqslant I_T(\mathbf{M}^*(t; H, \bar{T}_m, \tilde{\lambda})), \quad \forall \quad H \tag{38}$$

In the second step, we focus on the minimization of the fuel cost by searching for $\tilde{\lambda}^* \in \tilde{\Lambda}$ such that

$$I_T(\mathbf{M}^*(t; H^*_{\tilde{\lambda}^*}, \bar{T}_m, \tilde{\lambda}^*)) \leqslant I_T(\mathbf{M}^*(t; H^*_{\tilde{\lambda}}, \bar{T}_m, \tilde{\lambda})), \quad \forall \quad \tilde{\lambda} \in \tilde{\Lambda} \tag{39}$$

Combining (38) and (39), we obtain

$$I_T(\mathrm{M}^*(t; H^*_{\tilde{\lambda}^*}, \bar{T}_m, \tilde{\lambda}^*)) \leqslant I_T(\mathrm{M}^*(t; H, \bar{T}_m, \tilde{\lambda})) \tag{40}$$

for all H and $\tilde{\lambda} \in \tilde{\Lambda}$. Thus, we obtain $H^*_{\tilde{\lambda}^*}$ and $\tilde{\lambda}^*$, which are conditionally optimal with $\bar{T}_m$, and $\mathrm{M}^*(t; H^*_{\tilde{\lambda}^*}, \bar{T}_m, \tilde{\lambda}^*)$ is the fuel optimal manoeuvre set, which accomplishes the spacecraft formation reconfiguration with minimal total fuel cost.

To further elaborate upon the preceding discussion, consider an n to m spacecraft formation reconfiguration process with the manoeuvre time interval $[t_a, t_b]$. First, for the new spacecraft formation pattern that is specified by a particular choice of $\tilde{\lambda}$, spacecraft i can generate the fuel optimal manoeuvre $\mathrm{m}^*_{i,j}(t; t_a, t_b, \tilde{\lambda})$ with the corresponding fuel cost $I(\mathrm{m}^*_{i,j}(t; t_a, t_b, \tilde{\lambda}))$ for $j = 1, \ldots, m$. With boundary conditions (7) and (8) satisfied at t_a and t_b, respectively, the generation of $\mathrm{m}^*_{i,j}(t; t_a, t_b, \tilde{\lambda})$ can be accomplished via linear/nonlinear programming or calculus of variation. In this paper, for the aforementioned optimization process, we apply the direct solution of Euler–Lagrange equations from the theory of calculus of variation. Since each spacecraft will only generate the fuel optimal manoeuvres for itself going to different allowable positions in the new formation pattern, this part of computation is distributed and implemented in parallel on different spacecraft. Next, a 'dynamic-programming-styled' task assignment algorithm and a communication protocol for information exchange among all the spacecraft is used to determine the conditionally optimal $H^*_{\tilde{\lambda}}$. A significant advantage of this dynamic-programming-styled task assignment algorithm is that we are guaranteed to obtain $H^*_{\tilde{\lambda}}$ without exhaustively computing $I_T(\mathrm{M}^*(t; H, \bar{T}_m, \tilde{\lambda}))$ for all $\mathscr{P}^m_n$ different spacecraft permutations H. Thus, with $H^*_{\tilde{\lambda}}$ for the particular choice of $\tilde{\lambda} \in \tilde{\Lambda}$, $I_T(\mathrm{M}^*(t; H^*_{\tilde{\lambda}}, \bar{T}_m, \tilde{\lambda}))$ is given by

$$I_T(\mathrm{M}^*(t; H^*_{\tilde{\lambda}}, \bar{T}_m, \tilde{\lambda})) = \sum_{j=1}^m I(\mathrm{m}^*_{h^*_{\tilde{\lambda}_j}, j}(t; t_a, t_b, \tilde{\lambda})) \tag{41}$$

Next, we repeat the above steps to obtain $I_T(\mathrm{M}^*(t; H^*_{\tilde{\lambda}}, \bar{T}_m, \tilde{\lambda}))$ for different $\tilde{\lambda} \in \tilde{\Lambda}$ so that we can find $\tilde{\lambda}^*$, for which $I_T(\mathrm{M}^*(t; H^*_{\tilde{\lambda}^*}, \bar{T}_m, \tilde{\lambda}^*)) \leqslant I_T(\mathrm{M}^*(t; H^*_{\tilde{\lambda}}, \bar{T}_m, \tilde{\lambda}))$, for all $\tilde{\lambda} \in \tilde{\Lambda}$. This is a black-box problem for input $\tilde{\lambda} \in \tilde{\Lambda}$ and output $I_T(\mathrm{M}^*(t; H^*_{\tilde{\lambda}}, \bar{T}_m, \tilde{\lambda}))$, since we have no knowledge of the explicit relationship between $\tilde{\lambda}$ and $I_T(\mathrm{M}^*(t; H^*_{\tilde{\lambda}}, \bar{T}_m, \tilde{\lambda}))$. Thus, for the optimization of fuel cost for formation reconfiguration with respect to $\tilde{\lambda}$, we utilize genetic algorithm (GA), which has been practically proven for its efficacy in black-box optimization. Specifically, GA generates different gene strings such that each gene string is decoded to a specific $\tilde{\lambda} \in \tilde{\Lambda}$. Then the fitness value for the gene string is obtained with consideration of $I_T(\mathrm{M}^*(t; H^*_{\tilde{\lambda}}, \bar{T}_m, \tilde{\lambda}))$, i.e. a higher fitness value corresponds to a lower $I_T(\mathrm{M}^*(t; H^*_{\tilde{\lambda}}, \bar{T}_m, \tilde{\lambda}))$. Next, genetic operations are performed on the gene strings according to their fitness values. After several generations of genetic evolution, the best gene string will eventually converge to the string that is decoded to $\tilde{\lambda}^*$. See Figure 5 for a flow chart representation of this optimization algorithm architecture.

Note that for the above two-step conditional optimization problem, we may select to first find the conditionally optimal $\tilde{\lambda}^*_H$ for a given H. Next, we may determine H^* and the corresponding $\tilde{\lambda}^*_{H^*}$ so that $I_T(\mathrm{M}^*(t; H^*, \bar{T}_m, \tilde{\lambda}^*_{H^*})) \leqslant I_T(\mathrm{M}^*(t; H, \bar{T}_m, \tilde{\lambda}))$ for all H and $\tilde{\lambda}$. In this approach, the search space for $\tilde{\lambda}$ is the same as the first approach. However, the search for $\tilde{\lambda}^*_H \in \tilde{\Lambda}$ has to be repeated exhaustively $\mathscr{P}^m_n$ times for every H. Thus, this approach requires a significantly larger computational effort than the first approach, for which the computational effort for the search in H is greatly reduced by the dynamic-programming-styled task assignment algorithm.
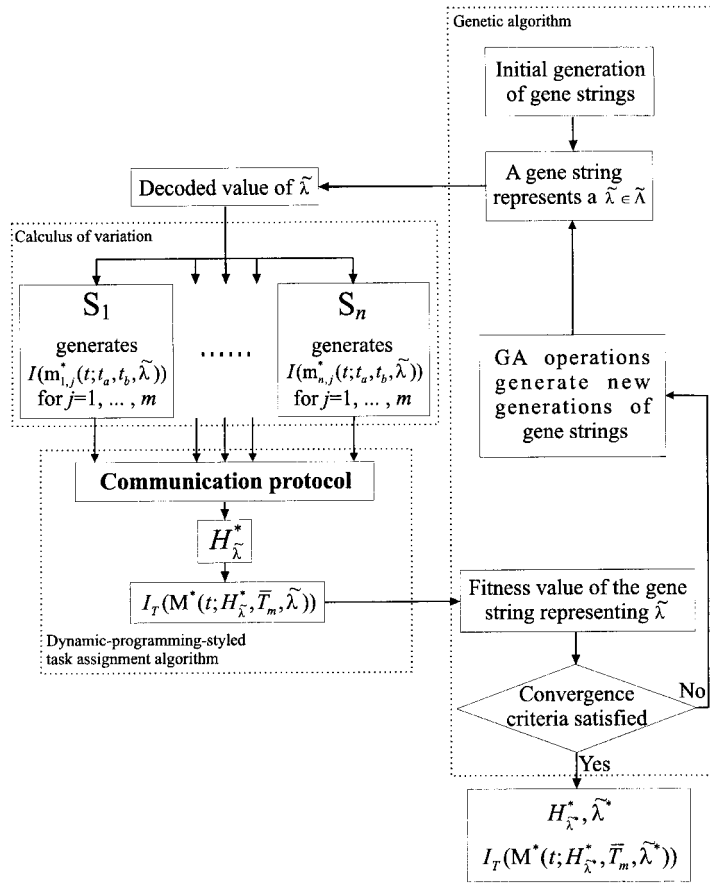
Figure 5. Optimization algorithm architecture.

4.2. Fuel optimal Manoeuvre for a single spacecraft

To generate the fuel optimal manoeuvre for a single spacecraft engaged in formation reconfiguration manoeuvre, we consider the linearized spacecraft relative motion dynamics characterized in the $\{x_I, y_I, z_I\}$ coordinate frame. The linearized spacecraft relative motion dynamics is typically described by the Hill's equations [19, 20, 22, 23, 26, 27] as below

$$\ddot{x}(t) = 2\omega\dot{y}(t) + u_x(t)$$

$$\ddot{y}(t) = -2\omega\dot{x}(t) + 3\omega^2 y(t) + u_y(t)$$

$$\ddot{z}(t) = -\omega^2 z(t) + u_z(t) \tag{42}$$

where $(x(t), y(t), z(t))$ is the spacecraft relative motion trajectory expressed in the $\{x_I, y_I, z_I\}$ coordinate frame and $u_x(t)$, $u_y(t)$, and $u_z(t)$ are the components of the thrust forces per unit mass along the x_I, y_I, and z_I axis, respectively.

If we prescribe a spacecraft relative motion trajectory $(x(t), y(t), z(t))$ in the $\{x_I, y_I, z_I\}$ coordinate frame, then the thrust force per unit mass $(u_x(t), u_y(t), u_z(t))$, which enables the spacecraft relative motion $(x(t), y(t), z(t))$, is given by

$$u_x(t) = \ddot{x}(t) - 2\omega\dot{y}(t)$$

$$u_y(t) = \ddot{y}(t) + 2\omega\dot{x}(t) - 3\omega^2 y(t)$$

$$u_z(t) = \ddot{z}(t) + \omega^2 z(t) \tag{43}$$

For notational convenience, in the balance of this subsection, we suppress the explicit dependence of various variables on t.

Let **M** be the mass of the spacecraft. Restricting consideration to a relatively short period of time, i.e. several spacecraft orbital periods out of several years long spacecraft life time, it follows that the mass of the spacecraft during a formation reconfiguration manoeuvre can be taken as a constant. If (u_x, u_y, u_z) is the thrust force per unit mass, the total thrust is $(\mathbf{M}\,u_x, \mathbf{M}\,u_y, \mathbf{M}\,u_z)$. The fuel consumption for a spacecraft to apply the thrust $(\mathbf{M}\,u_x, \mathbf{M}\,u_y, \mathbf{M}\,u_z)$ is governed by the type of the spacecraft propulsion system on-board. Two common spacecraft propulsion systems are the constant specific impulse propulsion system and the variable specific impulse (VSI) propulsion system. Of these, the VSI propulsion system, such as the electric plasma thruster, is particularly relevant to spacecraft orbit control and reconfiguration manoeuvre. Thus, in this paper, we assume that all spacecraft engaged in formation reconfiguration utilize the VSI propulsion system.

Let $\dot{\mathrm{m}}$ be the fuel mass consumption rate. Then, for a VSI propulsion system, $\dot{\mathrm{m}}$ is given by [35]

$$\dot{\mathrm{m}} = \frac{\mathbf{M}^2}{2P}(u_x^2 + u_y^2 + u_z^2) \tag{44}$$

where P is the power delivered to the propulsion system. For a time interval $[t_a, t_b]$, the total fuel mass consumption of a spacecraft is $\int_{ta}^{tb} \dot{\mathrm{m}}\,\mathrm{d}t$. Next, let the fuel cost for spacecraft i to execute a manoeuvre $\mathrm{m}_{i,j}(t; t_a, t_b, \tilde{\lambda})$ to relocate to the jth allowable position in the new formation pattern that is specified by $\tilde{\lambda}$ be defined as the total fuel mass consumption for this manoeuvre.

Next, we use Ψ to denote the set of all space curves (x, y, z), where x, y, and z are functions defined on $[t_a, t_b]$ that have continuous second derivatives on $[t_a, t_b]$ and satisfy the boundary conditions (7) and (8) at $t = t_a$ and $t = t_b$, respectively. Thus, every $(x, y, z) \in \Psi$ is a feasible spacecraft manoeuvre for spacecraft i to transfer to the jth allowable position in the new formation pattern that is specified by $\tilde{\lambda}$, starting at t_a and ending at t_b. In particular, x, y, and z represent simplified notations for $x_{\mathrm{m}_{i,j}}(t; \tilde{\lambda})$, $y_{\mathrm{m}_{i,j}}(t; \tilde{\lambda})$, and $z_{\mathrm{m}_{i,j}}(t; \tilde{\lambda})$, respectively. Now eliminating (u_x, u_y, u_z) in (44) using (43), we obtain the fuel cost for spacecraft i to execute manoeuvre $\mathrm{m}_{i,j}(t; t_a, t_b, \tilde{\lambda})$ as follows:

$$I(\mathrm{m}_{i,j}(t; t_a, t_b, \tilde{\lambda})) \overset{\triangle}{=} \int_{t_a}^{t_b} \dot{\mathrm{m}}\,\mathrm{d}t$$

$$= \frac{\mathbf{M}^2}{2P} \int_{t_a}^{t_b} ((\ddot{x} - 2\omega\dot{y})^2 + (\ddot{y} + 2\omega\dot{x} - 3\omega^2 y)^2 + (\ddot{z} + \omega^2 z)^2)\,\mathrm{d}t \tag{45}$$

For notational convenience, we now define

$$f(x, \dot{x}, \ddot{x}, y, \dot{y}, \ddot{y}, z, \dot{z}, \ddot{z}) \triangleq (\ddot{x} - 2\omega\dot{y})^2 + (\ddot{y} + 2\omega\dot{x} - 3\omega^2 y)^2 + (\ddot{z} + \omega^2 z)^2, \quad (x, y, z) \in \Psi \quad (46)$$

$$J(x, y, z) \triangleq \int_{t_a}^{t_b} f(x, \dot{x}, \ddot{x}, y, \dot{y}, \ddot{y}, z, \dot{z}, \ddot{z}) \, \mathrm{d}t, \quad (x, y, z) \in \Psi \quad (47)$$

Then, using (45)–(47), the fuel cost for spacecraft i to execute manoeuvre $\mathrm{m}_{i,j}(t; t_a, t_b, \tilde{\lambda})$ is given by

$$I(\mathrm{m}_{i,j}(t; t_a, t_b, \tilde{\lambda})) = \frac{\mathbf{M}^2}{2P} J(x, y, z), \quad (x, y, z) \in \Psi \quad (48)$$

To find the optimal manoeuvre $\mathrm{m}_{i,j}^*(t; t_a, t_b, \tilde{\lambda})$ for spacecraft i such that its fuel cost $I(\mathrm{m}_{i,j}^*(t; t_a, t_b, \tilde{\lambda}))$ is minimal, we search for (x^*, y^*, z^*) in Ψ such that $J(x^*, y^*, z^*)$ is minimal. This fuel optimization problem is addressed via the theory of calculus of variation. In particular, using the theory of calculus of variation, we can obtain an extremum that satisfies the Euler–Lagrange equations and boundary conditions (7) and (8) at $t = t_a$ and $t = t_b$, respectively. Note that since the set of space curves Ψ defined above is convex, the existence and uniqueness of $\min_{(x,y,z)\in\Psi} J(x, y, z)$ follows from the fact that $J(x, y, z)$ in (47) is a strictly convex functional on Ψ [36]. Thus, it follows that the extremum yields the unique optimal solution (x^*, y^*, z^*) for which $J(x^*, y^*, z^*)$ is minimal.

Next, the Euler–Lagrange equations for the minimization of (47) are given by [37, 38]

$$\frac{\partial f}{\partial q} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial f}{\partial \dot{q}}\right) + \frac{\mathrm{d}^2}{\mathrm{d}t^2}\left(\frac{\partial f}{\partial \ddot{q}}\right) = 0, \quad q \in \{x, y, z\} \quad (49)$$

where, for notational simplicity, we have suppressed the arguments of f (see (46) for details). Now, using (46) in (49), we obtain

$$\frac{\mathrm{d}^4 x}{\mathrm{d}t^4} - 4\omega\frac{\mathrm{d}^3 y}{\mathrm{d}t^3} - 4\omega^2\frac{\mathrm{d}^2 x}{\mathrm{d}t^2} + 6\omega^3\frac{\mathrm{d}y}{\mathrm{d}t} = 0$$

$$\frac{\mathrm{d}^4 y}{\mathrm{d}t^4} + 4\omega\frac{\mathrm{d}^3 x}{\mathrm{d}t^3} - 10\omega^2\frac{\mathrm{d}^2 y}{\mathrm{d}t^2} - 6\omega^3\frac{\mathrm{d}x}{\mathrm{d}t} + 9\omega^4 y = 0$$

$$\frac{\mathrm{d}^4 z}{\mathrm{d}t^4} + 2\omega^2\frac{\mathrm{d}^2 z}{\mathrm{d}t^2} + \omega^4 z = 0 \quad (50)$$

The general solution of the set of linear ordinary differential equations (50) is given by

$$x(t) = c_1 + c_2 t + c_3 t^2 + c_4 t^3 + c_5 \sin(\omega t) + c_6 \cos(\omega t) + c_7 t\cos(\omega t) + c_8 t\sin(\omega t)$$

$$y(t) = \left(\frac{2c_2}{3\omega} + \frac{16c_4}{9\omega^3}\right) + \frac{4c_3}{3\omega}t + \frac{2c_4}{\omega}t^2 + \left(\frac{3c_7}{10\omega} + \frac{c_5}{2}\right)\cos(\omega t) + \left(\frac{3c_8}{10\omega} - \frac{c_6}{2}\right)\sin(\omega t)$$

$$+ \frac{c_8}{2}t\cos(\omega t) - \frac{c_7}{2}t\sin(\omega t)$$

$$z(t) = c_9\sin(\omega t) + c_{10}\cos(\omega t) + c_{11}t\cos(\omega t) + c_{12}t\sin(\omega t) \quad (51)$$

where $t \in [t_a, t_b]$ and c_i, $i = 1, \ldots, 12$, are 12 integration constants, which can be determined using the boundary conditions (7) and (8) at $t = t_a$ and $t = t_b$, respectively. Since (50) is a

time-invariant linear system, we can shift the time variable from t to $\tau \overset{\triangle}{=} t - t_a$ in (51) to obtain an equivalent characterization of the general solution of (50) given by

$$x(\tau) = c_1 + c_2\tau + c_3\tau^2 + c_4\tau^3 + c_5\sin(\omega\tau) + c_6\cos(\omega\tau) + c_7\tau\cos(\omega\tau) + c_8\tau\sin(\omega\tau)$$

$$y(\tau) = \left(\frac{2c_2}{3\omega} + \frac{16c_4}{9\omega^3}\right) + \frac{4c_3}{3\omega}\tau + \frac{2c_4}{\omega}\tau^2 + \left(\frac{3c_7}{10\omega} + \frac{c_5}{2}\right)\cos(\omega\tau) + \left(\frac{3c_8}{10\omega} - \frac{c_6}{2}\right)\sin(\omega\tau)$$

$$\qquad + \frac{c_8}{2}\tau\cos(\omega\tau) - \frac{c_7}{2}\tau\sin(\omega\tau)$$

$$z(\tau) = c_9\sin(\omega\tau) + c_{10}\cos(\omega\tau) + c_{11}\tau\cos(\omega\tau) + c_{12}\tau\sin(\omega\tau) \qquad (52)$$

where $\tau \in [0, \Delta]$, with $\Delta \overset{\triangle}{=} t_b - t_a$.

With the general solution of (50) characterized by (52), an application of the boundary conditions (7) and (8) at $t = t_a$ (equivalently, at $\tau = 0$) and $t = t_b$ (equivalently, at $\tau = \Delta$), respectively, yields a linear system of 12 algebraic equations for c_i, $i = 1, \ldots, 12$. Having solved the linear system of 12 algebraic equations for c_i, $i = 1, \ldots, 12$, the 12 integration constants are substituted in (52) to produce the extremum that is the unique optimal (x^*, y^*, z^*), for all $(x, y, z) \in \Psi$. Thus, $J(x^*, y^*, z^*)$ is minimal, i.e. $J(x^*, y^*, z^*) < J(x, y, z)$ for any other $(x, y, z) \in \Psi$. Now the unique fuel optimal manoeuvre $\mathrm{m}_{i,j}^*(t; t_a, t_b, \tilde{\lambda}) = (x_{\mathrm{m}_{i,j}^*}(t; \tilde{\lambda}), y_{\mathrm{m}_{i,j}^*}(t; \tilde{\lambda}), z_{\mathrm{m}_{i,j}^*}(t; \tilde{\lambda}))$, $t_a \leqslant t \leqslant t_b$, is obtained from $x_{\mathrm{m}_{i,j}^*}(t; \tilde{\lambda}) = x^*$, $y_{\mathrm{m}_{i,j}^*}(t; \tilde{\lambda}) = y^*$, and $z_{\mathrm{m}_{i,j}^*}(t; \tilde{\lambda}) = z^*$, which guarantees that $I(\mathrm{m}_{i,j}^*(t; t_a, t_b, \tilde{\lambda})) < I(\mathrm{m}_{i,j}(t; t_a, t_b, \tilde{\lambda}))$ for any other $\mathrm{m}_{i,j}(t; t_a, t_b, \tilde{\lambda})$.

Finally, with the change of variable $\tau \overset{\triangle}{=} t - t_a$ and substitution of (46) and (52), the integral of (47) yields

$$\begin{aligned}
J(x^*, y^*, z^*) =& \left(\frac{4}{9}c_3^2 + \frac{16}{9\omega^2}c_4^2 + \frac{\omega^2}{10}(c_7^2 + c_8^2) + 2\omega^2(c_{11}^2 + c_{12}^2)\right)\Delta + \frac{4c_3c_4}{3}\Delta^2 + \frac{4c_4^2}{3}\Delta^3 \\
&+ \left(\frac{32c_7c_4}{15\omega} - \frac{8c_8c_3}{15}\right)\sin(\omega\Delta) - \left(\frac{32c_8c_4}{15\omega} + \frac{8c_7c_3}{15}\right)\cos(\omega\Delta) - \frac{8c_8c_4}{5}\Delta\sin(\omega\Delta) \\
&- \frac{8c_7c_4}{5}\Delta\cos(\omega\Delta) + \left(\frac{3\omega c_7c_8}{25} + 4\omega c_{11}c_{12}\right)\cos^2(\omega\Delta) \\
&+ \left(\frac{3\omega}{50}(c_8^2 - c_7^2) + 2\omega(c_{12}^2 - c_{11}^2)\right)\cos(\omega\Delta)\sin(\omega\Delta) \\
&+ \frac{8c_7c_3}{15} - \frac{3\omega c_7c_8}{25} + \frac{32c_8c_4}{15\omega} - 4\omega c_{11}c_{12} \qquad (53)
\end{aligned}$$

where, as elaborated above, the values of c_i, $i = 1, \ldots, 12$, correspond to the solution of the linear system of 12 algebraic equations consistent with the boundary conditions (7) and (8) at $t = t_a$ and $t = t_b$, respectively. Finally, the value of $I(\mathrm{m}_{i,j}^*(t; t_a, t_b, \tilde{\lambda}))$ is given by

$$I(\mathrm{m}_{i,j}^*(t; t_a, t_b, \tilde{\lambda})) = \frac{\mathbf{M}^2}{2P}J(x^*, y^*, z^*) \qquad (54)$$

4.3. Dynamic-programming-styled multi-agent task assignment algorithm

4.3.1. Task assignment problem

We begin by considering a general task assignment problem, where m different tasks are to be assigned to n agents, with $m \leqslant n$. Let the minimum cost for agent i to perform task j be denoted

as $C_{i,j}$. In addition, let the full decisions for all task assignments be denoted by an m dimensional *full decision vector* $[d_1, \ldots, d_m]$, where $d_j \in \{1, \ldots, n\}$, $j = 1, \ldots, m$, and $d_i \neq d_j$, for $i \neq j$. Specifically, d_j is the decision that agent d_j performs task j. Then, the cost for decision d_j is $C_{d_j,j}$. When only j, $1 \leq j < m$, decisions have been made, we denote an m dimensional *intermediate decision vector* as $[d_1, \ldots, d_j, 0_{m-j}]$, $j = 1, \ldots, m-1$. An optimal full decision vector is denoted as $[d_1^*, \ldots, d_m^*]$ and satisfies

$$\sum_{j=1}^{m} C_{d_j^*,j} \leqslant \sum_{j=1}^{m} C_{d_j,j} \tag{55}$$

for all full decision vectors $[d_1, \ldots, d_m]$. Note that the optimal task assignment problem is a multistage (actually, an m-stage) decision process, which can be solved efficiently using the well established dynamic programming methodology.

Next, we assign a state variable a_i to agent i, $i = 1, \ldots, n$, where $a_i \in \{0, 1\}$. The two values of the state variables 0 and 1 represent whether an agent is standing by for a task assignment or whether it has been assigned to perform a certain task, respectively. Thus, an n dimensional *state vector* $\tilde{a}_j \triangleq [a_1, \ldots, a_n]$ denotes the state of the n-agent system at the jth decision stage, $j = 0, \ldots, m$. For example, $\tilde{a}_0 = [0, \ldots, 0]$ denotes the state of the n-agent system at the '0' stage, where all agents are standing by. If $m = n$, $\tilde{a}_n = [1, \ldots, 1]$ denotes the state of the n-agent system at the final decision stage, where all agents have been assigned with certain tasks. The multistage task assignment process begins with the initial state $\tilde{a}_0$. At the jth stage, the jth decision (viz., d_j) is made and the intermediate decision vector is changed from $[d_1, \ldots, d_{j-1}, 0_{m-(j-1)}]$ to $[d_1, \ldots, d_{j-1}, d_j, 0_{m-j}]$. That is, agent d_j, which was standing by, is now assigned to perform task j and the value of a_{d_j} is changed from 0 to 1. Specifically, the full state of the n-agent system is changed from $\tilde{a}_{j-1}$ to $\tilde{a}_j$, where the d_jth component of $\tilde{a}_{j-1}$ is changed from 0 to 1. Let $\tilde{e}_i$ be the ith row of an $n \times n$ identity matrix. Then, the transformation from $\tilde{a}_{j-1}$ to $\tilde{a}_j$ with decision d_j can be performed by using an operator $\mathscr{D}$ as follows

$$\tilde{a}_j = \mathscr{D}(\tilde{a}_{j-1}, d_j)$$
$$\triangleq \tilde{a}_{j-1} + \tilde{e}_{d_j} \tag{56}$$

We define the inverse operator of $\mathscr{D}$ as

$$\tilde{a}_{j-1} = \mathscr{D}^{-1}(\tilde{a}_j, d_j)$$
$$\triangleq \tilde{a}_j - \tilde{e}_{d_j} \tag{57}$$

Note that after the jth decision, the state of the n-agent system $\tilde{a}_j$ has j non-zero components since each decision changes one component of $[a_1, \ldots, a_n]$ from 0 to 1. Thus, there are $\mathscr{C}_n^j$ possible states for the n-agent system at the jth stage, where C is the combination operator defined as $\mathscr{C}_n^j \triangleq n!/j!(n-j)!$ [34]. We let $\tilde{a}_{j_p}$, $p = 1, \ldots, \mathscr{C}_n^j$, denote one of these possible states for the n-agent system at the jth stage, where $\tilde{a}_{j_p}$ has j non-zero components.

Now we define $F_j(\tilde{a}_{j_p})$ as the minimum cost for the n-agent system to move from $\tilde{a}_0$ to a possible state $\tilde{a}_{j_p}$, $p = 1, \ldots, \mathscr{C}_n^j$, at the jth stage according to some intermediate decision vector $[d_{1_p}, \ldots, d_{j_p}, 0_{m-j}]$ for the first j decisions, i.e.

$$F_j(\tilde{a}_{j_p}) \triangleq \min_{[d_{1_p}, \ldots, d_{j_p}, 0_{m-j}]} \sum_{l=1}^{j} C_{d_{l_p},l} \tag{58}$$

where $[d_{1_p}, \ldots, d_{j_p}, 0_{m-j}]$ is any feasible intermediate decision vector that satisfies

$$\mathscr{D}(\ldots \mathscr{D}(\mathscr{D}(\tilde{a}_0, d_{1_p}), d_{2_p}), \ldots, d_{j_p}) = \sum_{l=1}^{j} \tilde{e}_{d_{l_p}}$$

$$= \tilde{a}_{j_p} \qquad (59)$$

Finally, since there are $\mathscr{C}_n^m$ possible states at the mth decision stage, we have

$$\sum_{j=1}^{m} C_{d_j^*, j} = \min_{p=1, \ldots, \mathscr{C}_n^m} F_m(\tilde{a}_{m_p}) \qquad (60)$$

where $F_m(\tilde{a}_{m_p})$ denotes the minimum cost for the n-agent system to move from $\tilde{a}_0$ to a possible state $\tilde{a}_{m_p}$, $p = 1, \ldots, \mathscr{C}_n^m$, at the mth stage according to some full decision vector $[d_{1_p}, \ldots, d_{m_p}]$.

Note that in the case where $m = n$, $\mathscr{C}_n^m = \mathscr{C}_n^n = 1$, i.e., there is only one possible state at the nth decision stage, which is $\tilde{a}_n$. Thus, we have,

$$\sum_{j=1}^{n} C_{d_j^*, j} = F_n(\tilde{a}_n) \qquad (61)$$

At the jth decision stage, for a particular $\tilde{a}_{j_p}$, we assume its j non-zero components are $k_{1_p}, \ldots, k_{j_p}$ components. Thus, the set of allowable values for decision d_{j_p} that brings the state to this $\tilde{a}_{j_p}$ at the jth stage is $K_{j_p} \triangleq \{k_{1_p}, \ldots, k_{j_p}\}$, since $k_{1_p}, \ldots, k_{j_p}$ are the only non-zero components in the state vector so far. Now, applying the Principle of Optimality [37–39], we obtain the following recurrence relation for $F_j(\tilde{a}_{j_p})$:

$$F_j(\tilde{a}_{j_p}) = \min_{d_{j_p} \in K_{j_p}} \{C_{d_{j_p}, j} + F_{j-1}(\mathscr{D}^{-1}(\tilde{a}_{j_p}, d_{j_p}))\}, \quad p = 1, \ldots, \mathscr{C}_n^j \qquad (62)$$

Next, we can construct a dynamic-programming-styled task assignment algorithm for all agents since each stage of this algorithm is based on the recurrence relation (62). In order to start the algorithm, each agent must generate the costs for itself to perform the m different tasks in advance. For the first stage, $F_1(\tilde{a}_{1_p}) = C_{p,1}$, $p = 1, \ldots, n$, since $\mathscr{C}_n^1 = n$. At the jth stage, for every possible state $\tilde{a}_{j_p}$, agent d_{j_p}, $d_{j_p} \in K_{j_p}$, needs to take the sum of $C_{d_{j_p}, j} + F_{j-1}(\mathscr{D}^{-1}(\tilde{a}_{j_p}, d_{j_p}))$. Note that $C_{d_{j_p}, j}$ is the cost of agent d_{j_p} to perform task j, which is known to agent d_{j_p}. Furthermore, $F_{j-1}(\mathscr{D}^{-1}(\tilde{a}_{j_p}, d_{j_p}))$ is the result obtained from the last stage for some state $\mathscr{D}^{-1}(\tilde{a}_{j_p}, d_{j_p})$, which has $j - 1$ non-zero components and whose d_{j_p} component is zero. Then the value of $C_{d_{j_p}, j} + F_{j-1}(\mathscr{D}^{-1}(\tilde{a}_{j_p}, d_{j_p}))$ is communicated to the other agents for comparison with $C_{d'_{j_p}, j} + F_{j-1}(\mathscr{D}^{-1}(\tilde{a}_{j_p}, d'_{j_p}))$, $d'_{j_p} \in K_{j_p}$, $d'_{j_p} \neq d_{j_p}$. In addition, the minimum value of $C_{d_{j_p}, j} + F_{j-1}(\mathscr{D}^{-1}(\tilde{a}_{j_p}, d_{j_p}))$, $d_{j_p} \in K_{j_p}$, is taken as $F_j(\tilde{a}_{j_p})$ and communicated to all agents for use at the next stage. After the mth stage, $F_m(\tilde{a}_{m_p})$, $p = 1, \ldots, \mathscr{C}_n^m$, is known to every agent. Thus, the agents can easily obtain the value of $\sum_{j=1}^{m} C_{d_j^*, j}$ and the corresponding optimal full decision vector $[d_1^*, \ldots, d_m^*]$ by choosing the minimum value of $F_m(\tilde{a}_{m_p})$ among all $\tilde{a}_{m_p}$, $p = 1, \ldots, \mathscr{C}_n^m$.

In order to show the efficacy of this dynamic-programming-styled task assignment algorithm, we consider the computational effort for this algorithm. At the jth stage, where $2 \leqslant j \leqslant m$, a possible state $\tilde{a}_{j_p}$ can be reached by a total of j allowable decision values that belong to the set K_{j_p}. Thus, in order to obtain $F_j(\tilde{a}_{j_p})$ for a particular $\tilde{a}_{j_p}$, the agents need to do j additions to obtain $C_{d_{j_p}, j} + F_{j-1}(\mathscr{D}^{-1}(\tilde{a}_{j_p}, d_{j_p}))$ for all j allowable values of $d_{j_p} \in K_{j_p}$ so that they can compare these values and find the minimum value of $C_{d_{j_p}, j} + F_{j-1}(\mathscr{D}^{-1}(\tilde{a}_{j_p}, d_{j_p}))$. For all $\mathscr{C}_n^j$ possible states at the jth stage, the agents have to do $j \mathscr{C}_n^j$ additions. Since no calculation is needed for the first

stage, for all m stages, the total effort will be $\sum_{j=2}^{m} j\,\mathscr{C}_n^j$ additions. Note that

$$\sum_{j=2}^{m} j\,\mathscr{C}_n^j = \sum_{j=2}^{m} j\,\frac{n!}{(n-j)!\,j!}$$

$$= \sum_{j=2}^{m} \frac{(n-1)!\,n}{(n-j)!\,(j-1)!}$$

$$= n\sum_{j=2}^{m} \mathscr{C}_{n-1}^{j-1} \tag{63}$$

For an n-agent-m-task problem, $m \leqslant n$, we denote the total number of additions that are needed for this algorithm as $E(n,m) \triangleq n\sum_{j=2}^{m} \mathscr{C}_{n-1}^{j-1}$. Specifically, for an n-agent-n-task problem, where $m = n$, the total number of additions is

$$E(n,n) = n\sum_{j=2}^{n} \mathscr{C}_{n-1}^{j-1}$$

$$= n(2^{n-1} - 1) \tag{64}$$

In contrast to the above, the exhaustive evaluation of $\sum_{j=1}^{m} C_{d_j,j}$ for all $\mathscr{P}_n^m$ different agent-task assignments requires at least $\hat{E}(n,m) \triangleq \mathscr{P}_n^m$ additions, if we assume that for a particular $[d_1, \ldots, d_m]$ the evaluation of $\sum_{j=1}^{m} C_{d_j,j}$ requires only one addition. Thus, for a 10-agent-10-task problem, $E(10,10) \approx 5.1 \times 10^3$ while $\hat{E}(10,10) \approx 3.6 \times 10^6$, and for a 15-agent-15-task problem, $E(15,15) \approx 2.45 \times 10^5$ while $\hat{E}(15,15) \approx 1.3 \times 10^{12}$.

Remark 4.1
A variety of alternative schemes exists to solve the task assignment problem. For example, linear programming (LP) techniques have been adapted to address the task assignment problem. Specifically, the task assignment problem is frequently interpreted and solved as a transportation problem [40, 41]. In addition, the Hungarian method is also used to address the task assignment problem [40]. The LP formulations of the task assignment problem can be characterized as efficient, low computational cost solutions that necessitate an iterative, centralized implementation. In contrast, the proposed dynamic-programming-styled task assignment algorithm is sequential and decentralized in nature, which requires only m stages for an n-agent-m-task problem. Frequently, auction-based algorithms are also used for the task assignment problem [31]. Unfortunately, task assignment using an auction-based algorithm may not even yield the optimal solution. In addition, the auction-based algorithms for the task assignment problem require a central computer processor, which accepts and evaluates bids and assigns tasks [31]. The dynamic-programming-styled task assignment algorithm of this paper eliminates the need for a centralized computer since it uses a distributed computational architecture along with a communication protocol (see Section 4.3.2) to generate the optimal task assignment policy.

Remark 4.2
Traditionally, the dynamic-programming-based solution techniques for multistage optimization problems have been known to suffer from the curse of dimensionality [39]. Specifically, a dynamic programming formulation of any multistage optimization problem involving several

state variables explodes in dimension when each state variable has large number of discretization levels. Fortunately, the task assignment problem considered above gives rise to state variables that have binary discretization levels (0 and 1). This results in a considerably simple problem structure, which is well suited for a dynamic-programming-styled solution methodology. Unfortunately, however, the computational cost of the proposed algorithm grows rapidly with the number of agents, thus making it potentially infeasible for task assignment with large number of agents.

4.3.2. Formation reconfiguration using multi-agent task assignment

Returning to our n to m spacecraft formation reconfiguration problem, $m \leqslant n$, n spacecraft are the n agents and manoeuvring to m allowable positions in the new formation pattern are the m different tasks. Letting spacecraft i manoeuvre to the jth allowable position corresponds to assigning task j to agent i. The cost for agent i to perform task j is the minimum fuel cost for spacecraft i to manoeuvre from its thrust-free trajectory in the original formation to the jth allowable position in the new formation pattern, i.e.

$$C_{i,j} = I(\mathrm{m}_{i,j}^*(t; t_a, t_b, \tilde{\lambda})) \tag{65}$$

Then, for the new formation pattern that is specified by $\tilde{\lambda}$, the optimal full decision vector for the corresponding n-agent-m-task problem is the conditionally optimal choice of H with the specified $\tilde{\lambda}$, i.e.

$$H_{\tilde{\lambda}}^* = [d_1^*, \dots, d_m^*] \tag{66}$$

To apply the dynamic-programming-styled task assignment algorithm to an n to m spacecraft formation reconfiguration process, we propose a communication protocol among all the spacecraft such that the algorithm can be implemented by spacecraft communication and the computational effort is distributed on each spacecraft equally.

Before the algorithm starts, each spacecraft generates the minimum fuel costs for itself to manoeuvre to m different allowable positions in the new formation pattern. Specifically, spacecraft i obtains in advance all the values of $I(\mathrm{m}_{i,j}^*(t; t_a, t_b, \tilde{\lambda}))$, $j = 1, \dots, m$, using the framework of Subsection 4.2.

In order to describe the dynamic-programming-styled task assignment algorithm and the proposed communication protocol in an illustrative manner, we consider the following example. A 4 spacecraft formation is required to reconfigure to a new formation pattern, where only 3 spacecraft are needed. Assume that the minimum fuel costs for spacecraft i, $i = 1, \dots, 4$, to manoeuvre to the jth, $j = 1, 2, 3$, allowable positions in the new formation pattern are known and are as given in Table I. Note that initially, each spacecraft only has the information of the corresponding row of data.

At the first stage, spacecraft i, $i = 1, \dots, 4$, broadcasts $F_1(\tilde{a}_{1_i}) = C_{i,1}$, the corresponding intermediate decision vector $[i, 0, 0]$, and the corresponding state vector $\tilde{a}_{1_i}$, whose ith component is 1. This set of data is obtained by all participating spacecraft (see Table II).

At the second stage, spacecraft i, $i = 1, \dots, 4$, checks the state vector $\tilde{a}_{1_p}$, $p = 1, \dots, 4$. If the ith component of a state vector $\tilde{a}_{1_p}$ is still zero, spacecraft i calculates $C_{i,2} + F_1(\tilde{a}_{1_p})$, updates the corresponding intermediate decision vector by adding $d_2 = i$, and updates the corresponding state vector $\tilde{a}_{1_p}$ by changing its ith component a_i from 0 to 1. Each spacecraft stores its own set of data temporarily (see Table III).

Table I. Formation reconfiguration fuel cost data.

$C_{i,j} = I(m_{i,j}^*(t; t_a, t_b, \tilde{\lambda}))$	$j = 1$	$j = 2$	$j = 3$
$i = 1$	6.1	3.2	2.2
$i = 2$	8.1	5.8	3.0
$i = 3$	6.5	7.4	8.1
$i = 4$	8.2	8.9	7.0

Table II. Data broadcast at stage 1.

Broadcast	$F_1(\tilde{a}_{1_p})$	$[d_1, 0, 0]$	$\tilde{a}_{1_p}$
Spacecraft 1	$F_1(\tilde{a}_{1_1}) = C_{1,1} = 6.1$	$[1,0,0]$	$\tilde{a}_{1_1} = [1, 0, 0, 0]$
Spacecraft 2	$F_1(\tilde{a}_{1_2}) = C_{2,1} = 8.1$	$[2,0,0]$	$\tilde{a}_{1_2} = [0, 1, 0, 0]$
Spacecraft 3	$F_1(\tilde{a}_{1_3}) = C_{3,1} = 6.5$	$[3,0,0]$	$\tilde{a}_{1_3} = [0, 0, 1, 0]$
Spacecraft 4	$F_1(\tilde{a}_{1_4}) = C_{4,1} = 8.2$	$[4,0,0]$	$\tilde{a}_{1_4} = [0, 0, 0, 1]$

Table III. Intermediate data set at stage 2.

	$C_{i,2} + F_1(\tilde{a}_{1_p})$	$[d_1, d_2, 0]$	$\mathscr{D}(\tilde{a}_{1_p}, d_2)$
Spacecraft 1	$C_{1,2} + F_1(\tilde{a}_{1_2}) = 11.3$	$[2, 1, 0]$	$[1,1,0,0]$
	$C_{1,2} + F_1(\tilde{a}_{1_3}) = 9.7$	$[3, 1, 0]$	$[1,0,1,0]$
	$C_{1,2} + F_1(\tilde{a}_{1_4}) = 11.4$	$[4, 1, 0]$	$[1,0,0,1]$
Spacecraft 2	$C_{2,2} + F_1(\tilde{a}_{1_1}) = 11.9$	$[1, 2, 0]$	$[1,1,0,0]$
	$C_{2,2} + F_1(\tilde{a}_{1_3}) = 12.3$	$[3, 2, 0]$	$[0,1,1,0]$
	$C_{2,2} + F_1(\tilde{a}_{1_4}) = 14.0$	$[4, 2, 0]$	$[0,1,0,1]$
Spacecraft 3	$C_{3,2} + F_1(\tilde{a}_{1_1}) = 13.5$	$[1, 3, 0]$	$[1,0,1,0]$
	$C_{3,2} + F_1(\tilde{a}_{1_2}) = 15.5$	$[2, 3, 0]$	$[0,1,1,0]$
	$C_{3,2} + F_1(\tilde{a}_{1_4}) = 15.6$	$[4, 3, 0]$	$[0,0,1,1]$
Spacecraft 4	$C_{4,2} + F_1(\tilde{a}_{1_1}) = 15.0$	$[1, 4, 0]$	$[1,0,0,1]$
	$C_{4,2} + F_1(\tilde{a}_{1_2}) = 17.0$	$[2, 4, 0]$	$[0,1,0,1]$
	$C_{4,2} + F_1(\tilde{a}_{1_3}) = 15.4$	$[3, 4, 0]$	$[0,0,1,1]$

Next, spacecraft 1 broadcasts $C_{1,2} + F_1(\tilde{a}_{1_2}) = 11.3$ with the corresponding state vector $[1,1,0,0]$, $C_{1,2} + F_1(\tilde{a}_{1_3}) = 9.7$ with the corresponding state vector $[1,0,1,0]$, and $C_{1,2} + F_1(\tilde{a}_{1_4}) = 11.4$ with the corresponding state vector $[1,0,0,1]$. When the other spacecraft receive this data, each of them checks whether there is a match for some state vectors with its own set of temporarily stored data. If there is a match for a particular state vector, a comparison is made between the corresponding values of $C_{i,2} + F_1(\tilde{a}_{1_p})$, where one of these is from the spacecraft's own set of temporarily stored data and the other is received from spacecraft 1. If the value from the spacecraft's own set of temporarily stored data is greater than or equal to the value that is received from spacecraft 1, the spacecraft deletes the stored value of $C_{i,2} + F_1(\tilde{a}_{1_p})$ and the

corresponding intermediate decision vector and the state vector; otherwise the corresponding data in spacecraft's own set of temporarily stored data is retained. For example, spacecraft 2 deletes $C_{2,2} + F_1(\tilde{a}_{1_1})$, the corresponding intermediate decision vector, and the state vector from its own set of temporarily stored data since $C_{2,2} + F_1(\tilde{a}_{1_1})$ is greater than $C_{1,2} + F_1(\tilde{a}_{1_2})$, which has the same corresponding state vector $[1, 1, 0, 0]$. Similarly, spacecraft 3 deletes $C_{3,2} + F_1(\tilde{a}_{1_1})$, its corresponding intermediate decision vector, and the state vector and spacecraft 4 deletes $C_{4,2} + F_1(\tilde{a}_{1_1})$, its corresponding intermediate decision vector, and the state vector from their own set of temporarily stored data. Then, spacecraft 2 broadcasts its remaining data and the same comparison is carried out on the other spacecraft, including spacecraft 1, when they receive the data from spacecraft 2. This process continues until spacecraft 4 has broadcast and comparison for data has been made on the other spacecraft. If one spacecraft has deleted all the stored data before its turn to broadcast, it simply skips the broadcast. After spacecraft 4 broadcasts, the final remaining values on the spacecraft are $F_2(\tilde{a}_{2_p})$, $p = 1, \ldots, 6$, since $\mathcal{C}_4^2 = 6$. The resulting data set is given in Table IV. At the end of this stage, $F_2(\tilde{a}_{2_p})$, $p = 1, \ldots, 6$, together with the corresponding intermediate decision vector and the state vector, are broadcast by its holder so that this set of data is obtained by all the other spacecraft.

At the third stage, spacecraft i, $i = 1, \ldots, 4$, checks the state vector $\tilde{a}_{2_p}$, $p = 1, \ldots, 6$. If the ith component of a state vector $\tilde{a}_{2_p}$ is still zero, spacecraft i calculates $C_{i,3} + F_2(\tilde{a}_{2_p})$, updates the corresponding intermediate decision vector by adding $d_3 = i$, and also updates the corresponding state vector $\tilde{a}_{2_p}$ by changing its ith component a_i from 0 to 1. Each spacecraft stores these data temporarily (see Table V).

Now beginning with spacecraft 1, every spacecraft broadcasts in turn while the same comparison operation as described for the second stage is carried on the other spacecraft. Finally, the remaining data are shown in Table VI and they are broadcasted from their holder to all the other spacecraft.

After the third stage, which is the last one for this example since $m = 3$, every spacecraft has obtained all $F_3(\tilde{a}_{3_p})$, $p = 1, \ldots, 4$ since $\mathcal{C}_4^3 = 4$, and the corresponding full decision vectors. Clearly, they choose $[3, 1, 2]$ as the optimal full decision vector since the corresponding

Table IV. Final data set at stage 2.

	$F_2(\tilde{a}_{2_p})$	$[d_1, d_2, 0]$	$\tilde{a}_{2_p}$
Spacecraft 1	$F_2(\tilde{a}_{2_1}) = C_{1,2} + F_1(\tilde{a}_{1_2}) = 11.3$	$[2, 1, 0]$	$\tilde{a}_{2_1} = [1, 1, 0, 0]$
	$F_2(\tilde{a}_{2_2}) = C_{1,2} + F_1(\tilde{a}_{1_3}) = 9.7$	$[3, 1, 0]$	$\tilde{a}_{2_2} = [1, 0, 1, 0]$
	$F_2(\tilde{a}_{2_3}) = C_{1,2} + F_1(\tilde{a}_{1_4}) = 11.4$	$[4, 1, 0]$	$\tilde{a}_{2_3} = [1, 0, 0, 1]$
Spacecraft 2	—	—	—
	$F_2(\tilde{a}_{2_4}) = C_{2,2} + F_1(\tilde{a}_{1_3}) = 12.3$	$[3, 2, 0]$	$\tilde{a}_{2_4} = [0, 1, 1, 0]$
	$F_2(\tilde{a}_{2_5}) = C_{2,2} + F_1(\tilde{a}_{1_4}) = 14.0$	$[4, 2, 0]$	$\tilde{a}_{2_5} = [0, 1, 0, 1]$
Spacecraft 3	—	—	—
	—	—	—
	—	—	—
Spacecraft 4	—	—	—
	—	—	—
	$F_2(\tilde{a}_{2_6}) = C_{4,2} + F_1(\tilde{a}_{1_3}) = 15.4$	$[3, 4, 0]$	$\tilde{a}_{2_6} = [0, 0, 1, 1]$

Table V. Intermediate datat set at stage 3.

	$C_{i,3} + F_2(\tilde{a}_{2_p})$	$[d_1, d_2, d_3]$	$\mathscr{D}(\tilde{a}_{2_p}, d_3)$
Spacecraft 1	$C_{1,3} + F_2(\tilde{a}_{2_4}) = 14.5$	[3, 2, 1]	[1,1,1,0]
	$C_{1,3} + F_2(\tilde{a}_{2_5}) = 16.2$	[4, 2, 1]	[1,1,0,1]
	$C_{1,3} + F_2(\tilde{a}_{2_6}) = 17.6$	[3, 4, 1]	[1,0,1,1]
Spacecraft 2	$C_{2,3} + F_2(\tilde{a}_{2_2}) = 12.7$	[3, 1, 2]	[1,1,1,0]
	$C_{2,3} + F_2(\tilde{a}_{2_3}) = 14.4$	[4, 1, 2]	[1,1,0,1]
	$C_{2,3} + F_2(\tilde{a}_{2_6}) = 18.4$	[3, 4, 2]	[0,1,1,1]
Spacecraft 3	$C_{3,3} + F_2(\tilde{a}_{2_1}) = 19.4$	[2, 1, 3]	[1,1,1,0]
	$C_{3,3} + F_2(\tilde{a}_{2_3}) = 19.5$	[4, 1, 3]	[1,0,1,1]
	$C_{3,3} + F_2(\tilde{a}_{2_5}) = 22.1$	[4, 2, 3]	[0,1,1,1]
Spacecraft 4	$C_{4,3} + F_2(\tilde{a}_{2_1}) = 18.3$	[2, 1, 4]	[1,1,0,1]
	$C_{4,3} + F_2(\tilde{a}_{2_2}) = 16.7$	[3, 1, 4]	[1,0,1,1]
	$C_{4,3} + F_2(\tilde{a}_{2_4}) = 19.3$	[3, 2, 4]	[0,1,1,1]

Table VI. Final data set at stage 3.

	$F_3(\tilde{a}_{3_p})$	$[d_1, d_2, d_3]$	$\tilde{a}_{3_p}$
Spacecraft 1	—	—	—
	—	—	—
	—	—	—
Spacecraft 2	$F_3(\tilde{a}_{3_1}) = C_{2,3} + F_2(\tilde{a}_{2_2}) = 12.7$	[3, 1, 2]	$\tilde{a}_{3_1} = [1, 1, 1, 0]$
	$F_3(\tilde{a}_{3_2}) = C_{2,3} + F_2(\tilde{a}_{2_3}) = 14.4$	[4, 1, 2]	$\tilde{a}_{3_2} = [1, 1, 0, 1]$
	$F_3(\tilde{a}_{3_3}) = C_{2,3} + F_2(\tilde{a}_{2_6}) = 18.4$	[3, 4, 2]	$\tilde{a}_{3_3} = [0, 1, 1, 1]$
Spacecraft 3	—	—	—
	—	—	—
	—	—	—
Spacecraft 4	—	—	—
	$F_3(\tilde{a}_{3_4}) = C_{4,3} + F_2(\tilde{a}_{2_2}) = 16.7$	[3, 1, 4]	$\tilde{a}_{3_4} = [1, 0, 1, 1]$
	—	—	—

$F_3(\tilde{a}_{3_1}) = 12.7$ is the minimum total fuel cost. Thus, by communicating with each other, all the spacecraft can reach an agreement for the optimal decision that spacecraft 3 manoeuvres to the 1st allowable position, spacecraft 1 manoeuvres to the 2nd allowable position, and spacecraft 2 manoeuvres to the 3rd allowable position in the new formation pattern while spacecraft 4 remains in its thrust-free trajectory in the original formation. If the new formation pattern is specified by $\tilde{\lambda}$, then $H_{\tilde{\lambda}}^* = [3, 1, 2]$. Finally, the total fuel cost for $I_T(\mathbf{M}^*(t; H_{\tilde{\lambda}}^*, \bar{T}_m, \tilde{\lambda}))$ is $I_T(\mathbf{M}^*(t; H_{\tilde{\lambda}}^*, \bar{T}_m, \tilde{\lambda})) = 12.7$.

Remark 4.3

In the spirit of Morton *et al.* [31], the minimization of the total fuel cost for formation reconfiguration subject to an equalized fuel use by the spacecraft in formation can be accomplished by modifying $C_{i,j}$, $i = 1, \ldots, n$ and $j = 1, \ldots, m$, in (65) to include a factor that weighs the remaining fuel on-board of the spacecraft S_i, $i = 1, \ldots, n$.

Remark 4.4

The aforementioned communication protocol is characterized by a distributed implementation of the assignment algorithm and may necessitate a large amount of data communication throughout the decision process. Generally, for an n-agent-n-task assignment problem, there are at most (i.e., worst case) $(n + 2)2^{n-1}$ floating-point data strings and $(n + 2)2^{n-1}$ integer data strings to be communicated among agents. In an alternative communication strategy, each agent can broadcast its own initial data set to all the other agents so that all agents obtain the full set of initial data (i.e. Table I). Next, all agents may execute the assignment algorithm of this subsection in parallel and independently. At the successful termination of the algorithm, all agents broadcast and cross check their results. The overall computational requirement of this implementation is n times larger (thus, potentially, the total computation time may be n times longer) than that of the aforementioned distributed implementation. However, this communication strategy significantly reduces the total amount of data communication to about $n^2 + n$ floating-point data strings and n integer data strings.

4.4. Genetic algorithm

Finally, we address the problem of the search for optimal $\tilde{\lambda}^* \in \tilde{\Lambda}$ such that $I_T(\mathrm{M}^*(t; H_{\tilde{\lambda}^*}^*, \bar{T}_m, \tilde{\lambda}^*))$ is smaller than (or at worst equal to) $I_T(\mathrm{M}^*(t; H_{\tilde{\lambda}}^*, \bar{T}_m, \tilde{\lambda}))$, which results from any other choice of $\tilde{\lambda} \in \tilde{\Lambda}$. Unfortunately, an explicit relationship between the input $\tilde{\lambda}$ and the output $I_T(\mathrm{M}^*(t; H_{\tilde{\lambda}}^*, \bar{T}_m, \tilde{\lambda}))$ is unknown for this optimization problem. This eliminates the possibility of addressing this problem using the calculus-based optimization techniques, which rely on the existence of the derivative of $I_T(\mathrm{M}^*(t; H_{\tilde{\lambda}}^*, \bar{T}_m, \tilde{\lambda}))$ with respect to $\tilde{\lambda}$.

Genetic algorithm (GA) has been widely proven for its efficacy in black-box optimization [42–46]. Thus, in this paper, we adopt GA to search for $\tilde{\lambda}^*$. We briefly note that GA is a search algorithm that uses random selection as a tool to guide a highly exploitative search through a coding of the parameter space. However, we emphasize that in GA randomized search does not necessarily imply directionless search. Specifically, GA is directed by the evolutionary rule of the nature in its search for the optimal. In the sequel, we briefly review the basic aspects of GA utilized in this paper, viz. binary coding, Roulette Wheel selection, two-point crossover, jump-creep mutation, elitism, and population re-initialization. See References [42–44] for a detailed exposition of GA.

In this paper, Λ_i, the domain of λ_i, $i = 1, \ldots, k$, is assumed to be a continuous bounded interval. We use a 32-bit binary coding to discretize Λ_i to ensure sufficient precision. Thus, a gene string of total $k \times 32$-bit length is the unique binary code representation for a particular $\tilde{\lambda} = [\lambda_1, \ldots, \lambda_k] \in \tilde{\Lambda}$. The $k \times 32$-bit gene string length is also called the parameter length or the chromosome length.

Let N denote the population size, which is the total number of gene strings in one generation. Then, we generate the first generation of N gene strings randomly such that their decoded

representations of $\tilde{\lambda}$ are uniformly distributed in $\tilde{\Lambda}$. Next, starting from the first generation of gene strings, GA continues the random selection of gene strings such that the gene strings of higher fitness values have a greater probability of selection. The selected gene strings produce new gene strings such that the good genetic information, which is likely to provide higher fitness values, is passed to the next generation whereas the bad genetic information, which causes lower fitness values, is filtered out after several generations of gene selection. Based on the survival-of-the-fittest mechanism, this evolutionary process drives the whole population to converge to the gene strings of high fitness values.

Specifically, to select good gene strings from a current generation, we define the fitness value $g(\xi_i)$ of a $k \times 32$-bit gene string ξ_i, $i = 1, \ldots, N$, as

$$g(\xi_i) \triangleq -I_T(\mathrm{M}^*(t; H^*_{\tilde{\lambda}_i}, \bar{T}_m, \tilde{\lambda}_i)), \quad i = 1, \ldots, N \tag{67}$$

where ξ_i is a $k \times 32$-bit coded representation of $\tilde{\lambda}_i$. This definition ensures that a higher fitness value corresponds to a lower $I_T(\mathrm{M}^*(t; H^*_{\tilde{\lambda}}, \bar{T}_m, \tilde{\lambda},))$.

Next, a Roulette Wheel selection scheme is used for selecting the parent gene strings from a current generation of gene strings. In particular, for one generation of gene strings, let $g_{\max}$ and $g_{\min}$ represent the maximum and minimum fitness values, respectively. Next, define

$$G_i \triangleq g(\xi_i) - g_{\min} + 0.01(g_{\max} - g_{\min}) \tag{68}$$

In addition, let $R_0 \triangleq 0$ and using (68) define

$$R_i \triangleq \sum_{j=1}^{i} G_j, \; i = 1, \ldots, N \tag{69}$$

Note that the length of the interval $[R_{(i-1)}, R_i]$, $i = 1, \ldots, N$, is linearly dependent on the fitness value $g(\xi_i)$, i.e. a greater $g(\xi_i)$ corresponds to a larger interval $[R_{(i-1)}, R_i]$. Next, we generate a number uniformly at random in the range $(0, R_N)$ and if it falls into the range $[R_{(i-1)}, R_i]$ then we select ξ_i as a parent.

Suppose we have obtained two parent gene strings, then a two-point crossover happens when $p_1 > p_c$, where p_1 is a randomly generated number according to uniform distribution on $[0, 1]$ and $p_c \in [0, 1]$ is computed adaptively using [47]

$$p_c = \begin{cases} \dfrac{g_{\max} - g(\xi'_i)}{g_{\max} - \bar{g}}, & \bar{g} \leqslant g(\xi'_i) \\ 1, & g(\xi'_i) < \bar{g} \end{cases} \tag{70}$$

where $g(\xi'_i)$ is the better fitness value from the two parent strings and $\bar{g}$ is the average fitness value for the whole generation. In the crossover operation, two integers l_1 and l_2, which are called crossover positions, are selected uniformly at random from $\{1, \ldots, k \times 32\}$. The bits between the two selected crossover positions are exchanged between the two parent strings to generate two new offsprings.

The mutation operation follows crossover if $p_2 > p_m$, where p_2 is a randomly generated number according to uniform distribution on $[0, 1]$ and $p_m \in [0, 1]$ is computed

adaptively using [47]

$$
p_m = \begin{cases} \dfrac{g_{\max} - g(\xi_i)}{g_{\max} - \bar{g}}, & \bar{g} \leqslant g(\xi_i) \\[2ex] 1, & g(\xi_i) < \bar{g} \end{cases}
\tag{71}
$$

where $g(\xi_i)$ is the fitness value of gene string ξ_i and $\bar{g}$ is the average fitness value for the whole generation. First, in jump mutation a bit is chosen uniformly at random from $\{1, \ldots, k \times 32\}$ and its state is changed from 0 to 1 or *vice versa*. Next, in creep mutation, we perturb $\tilde{\lambda}$ corresponding to a gene string by a small amount $\delta\tilde{\lambda}$ such that $(\tilde{\lambda} + \delta\tilde{\lambda}) \in \tilde{\Lambda}$, and then code $(\tilde{\lambda} + \delta\tilde{\lambda})$ to a new gene string. This bit-wise jump mutation followed by creep mutation has showed satisfactory result in our simulation.

In each generation, we leave the gene string having the best fitness value unchanged and propagate it to the next generation. This operation, which is termed as elitism, guarantees non-decreasing best fitness value in successive generations.

To ensure fast monotone increase of the best fitness value, we use the population re-initialization when the search is considered to be trapped in a local minimum. Specifically, the best fitness value of each generation is recorded, and the standard deviation (σ) of the best fitness value of the latest s generations, $\sigma(s)$, is used as the criterion for whether or not GA search is trapped in a local minimum. We choose two positive integers s_1 and s_2, $s_1 < s_2$, and a small positive real number ε. If $\sigma(s_1) < \varepsilon$, we reinitialize the population randomly in the neighborhood of the current best individual to accelerate the GA search. If $\sigma(s_2) < \varepsilon$, we consider the GA search to be trapped in a local minimum and reinitialize the gene strings randomly in the whole search space $\tilde{\Lambda}$ while the gene string with the best fitness value is kept. If the population has been reinitialized in the whole search space for several times while there is no further improvement in the best fitness value among the population, we consider the highest fitness value has been reached. Then, the decoded value of best gene string, which has the highest fitness value, is considered to be $\tilde{\lambda}^*$.

5. ILLUSTRATIVE NUMERICAL EXAMPLES

In this section, we provide two illustrative numerical examples to demonstrate the proposed fuel optimal multiple spacecraft formation reconfiguration framework. The problem data for simulating the fuel optimal multiple spacecraft formation reconfiguration algorithm is adopted from Reference [9]. Specifically, each spacecraft in the initial formation is assumed to have an equal mass $\mathbf{M} = 77\,\mathrm{kg}$ and a total power input $P = 100\,\mathrm{W}$ for its VSI propulsion system, which is assumed to have 10 percent efficiency. An imaginary leader spacecraft is assumed to be in a Keplerian circular Earth orbit with radius 7178 km, which corresponds to an orbital period $T_\mathrm{I} = 1.681\,\mathrm{h}$. For each numerical example, the length of the formation reconfiguration manoeuvre time interval $[t_a, t_b]$ is assumed to be T_I, i.e. $t_b = t_a + T_\mathrm{I}$. Finally, for GA-based parameter optimization, we used the population size $N = 50$.

Example 5.1
In this example, we assume that initially one spacecraft is coasting near by a cluster of 5 spacecraft as shown in Figure 6(a). The initial spacecraft formation pattern consists of three primary formation orbits with the initial thrust free periodic trajectories of spacecraft $i =$
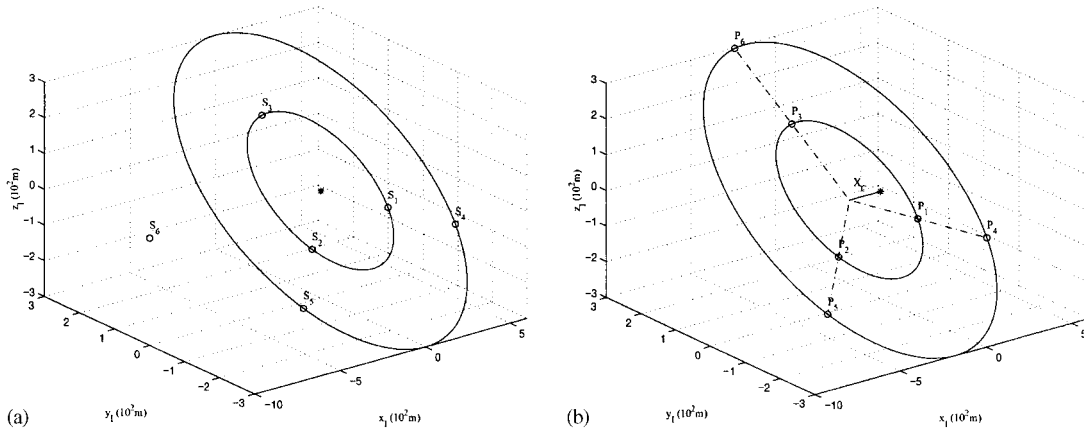
Figure 6. (a) Formation pattern at t_a; (b) desired formation pattern: Example 1 (★ denotes the imaginary leader spacecraft).

$1, \ldots, 6$ given by

$$x_{S_i}(t) = -2\frac{v_{y_0}}{\omega}\cos\left(\omega(t - t_a) + \theta + \beta_i\right) + X_C$$

$$y_{S_i}(t) = \frac{v_{y_0}}{\omega}\sin\left(\omega(t - t_a) + \theta + \beta_i\right)$$

$$z_{S_i}(t) = z_0\cos\left(\omega(t - t_a) + \theta + \beta_i\right) + \frac{v_{z_0}}{\omega}\sin\left(\omega(t - t_a) + \theta + \beta_i\right) \tag{72}$$

where $\omega = 1.038 \times 10^{-3}$ rad/s is the orbital angular velocity of the imaginary space-craft and $\theta = \pi/8$. In particular, spacecraft 1, 2, and 3 are distributed on the primary formation orbit specified by (72) with $(X_C, v_{y_0}, v_{z_0}, z_0) = (0, -150\,\omega\,\text{m/s}, -150\,\omega\,\text{m/s}, 0)$. Furthermore, for spacecraft 1, 2, and 3 we are given $\beta_1 = 0$, $\beta_2 = 2\pi/3$, and $\beta_3 = 4\pi/3$, respectively. Similarly, spacecraft 4 and 5 are distributed on the primary formation orbit specified by (72) with $(X_C, v_{y_0}, v_{z_0}, z_0) = (0, -300\,\omega\,\text{m/s}, -300\,\omega\,\text{m/s}, 0)$. Furthermore, for spacecraft 4 and 5 we are given $\beta_4 = 0$ and $\beta_5 = 2\pi/3$, res-pectively. Finally, spacecraft 6, which is coasting near by the cluster of 5 spacecraft, is on the primary formation orbit specified by (72) with $(X_C, v_{y_0}, v_{z_0}, z_0) = (-1000\,\text{m}, 0, 0, 0)$ and $\beta_6 = 0$, which reduces to a point at $(-1000\,\text{m}, 0, 0)$ in the $\{x_I, y_I, z_I\}$ coordinate frame.

Next, the 6 spacecraft are required to combine into one cluster so that a Y-shaped formation pattern can be formed as shown in Figure 6(b), where X_C is yet to be determined. Specifically, the allowable positions in the new formation pattern are distributed on two primary formation

orbits specified by

$$x_{P_i}(t) = -2\frac{v_{y_0}}{\omega}\cos\left(\omega(t-t_b)+\theta_P+\beta_i\right)+X_C$$

$$y_{P_i}(t) = \frac{v_{y_0}}{\omega}\sin\left(\omega(t-t_b)+\theta_P+\beta_i\right)$$

$$z_{P_i}(t) = z_0\cos\left(\omega(t-t_b)+\theta_P+\beta_i\right)+\frac{v_{z_0}}{\omega}\sin\left(\omega(t-t_b)+\theta_P+\beta_i\right) \quad (73)$$

In particular, the allowable positions 1, 2, and 3 in the new formation pattern are distributed on the primary formation orbit specified by (73) with $(X_C, v_{y_0}, v_{z_0}, z_0) = (X_C, -150\,\omega\,\text{m/s}, -150\,\omega\,\text{m/s}, 0)$. Furthermore, for the allowable positions 1, 2, and 3 in the new formation pattern $\beta_1 = 0$, $\beta_2 = 2\pi/3$, and $\beta_3 = 4\pi/3$, respectively. Similarly, the allowable positions 4, 5, and 6 in the new formation pattern are distributed on the primary formation orbit specified by (73) with $(X_C, v_{y_0}, v_{z_0}, z_0) = (X_C, -300\,\omega\,\text{m/s}, -300\,\omega\,\text{m/s}, 0)$. Furthermore, for the allowable positions 4, 5, and 6 in the new formation pattern $\beta_4 = 0$, $\beta_5 = 2\pi/3$, and $\beta_6 = 4\pi/3$, respectively. Note that in (73) X_C and θ_P are allowed to be selected freely from $[-1000, 1000]$ m and $[0, 2\pi/3]$ rad, respectively.

The optimization algorithm described in Section 4 yields $X_C^* = -182.212$ m and $\theta_P^* = 0.423$ rad for the new formation pattern. Note that the value of X_C^* indicates that the center of the initial cluster of 5 spacecraft moves from the origin of the $\{x_I, y_I, z_I\}$ coordinate frame, which is the location of the imaginary leader spacecraft, closer to the initial location of spacecraft 6 so as to form the new formation pattern at an intermediate location. Thus, the fuel burden of spacecraft 6 is shared by the other 5 spacecraft even as the total fuel consumption is minimized. The fuel cost for spacecraft S_i, $i = 1, \ldots, 6$, to relocate to the allowable location P_j, $j = 1, \ldots, 6$, in the new formation pattern specified by X_C^* and θ_P^* is given in Table VII. The dynamic-programming-styled task assignment algorithm yields $H^* = [1, 2, 6, 4, 5, 3]$ to be the optimal spacecraft permutation in the new formation pattern specified by X_C^* and θ_P^*. It is interesting to observe that the fuel optimal multiple spacecraft formation reconfiguration algorithm relocates spacecraft 3, which was initially on the inner primary formation orbit, to the 6th allowable position in the new formation pattern, which is on the outer primary formation orbit. In addition, spacecraft 6 joins the spacecraft cluster at the 3rd allowable position in the new formation pattern, which is on the inner primary formation orbit, instead of relocating to a vacant position in the initial formation pattern, which was on the outer primary formation orbit. Figure 7 gives the fuel optimal spacecraft formation reconfiguration trajectories for all the 6 spacecraft. Furthermore, Figure 8 provides the thrust histories for all the 6 spacecraft. Finally, the total fuel mass consumed by each spacecraft is listed in Table VIII.

Table VII. Formation reconfiguration fuel cost data: Example 1 (unit: 10^{-3} kg).

$I(m_{i,j}^*(t; t_a, t_b, \tilde{\lambda}^*))$	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$	$j=6$
$i=1$	0.0896	9.2394	8.4961	3.5600	21.7365	19.7880
$i=2$	10.1422	0.0875	9.2134	23.1259	2.8933	20.6833
$i=3$	9.5552	8.4114	0.1051	22.4091	19.9984	2.9238
$i=4$	2.9218	21.9044	20.3924	0.0933	37.9353	34.4493
$i=5$	22.8674	3.4408	21.6673	39.0655	0.0892	36.0802
$i=6$	3.5094	6.2365	4.8546	11.4744	16.8056	13.5798
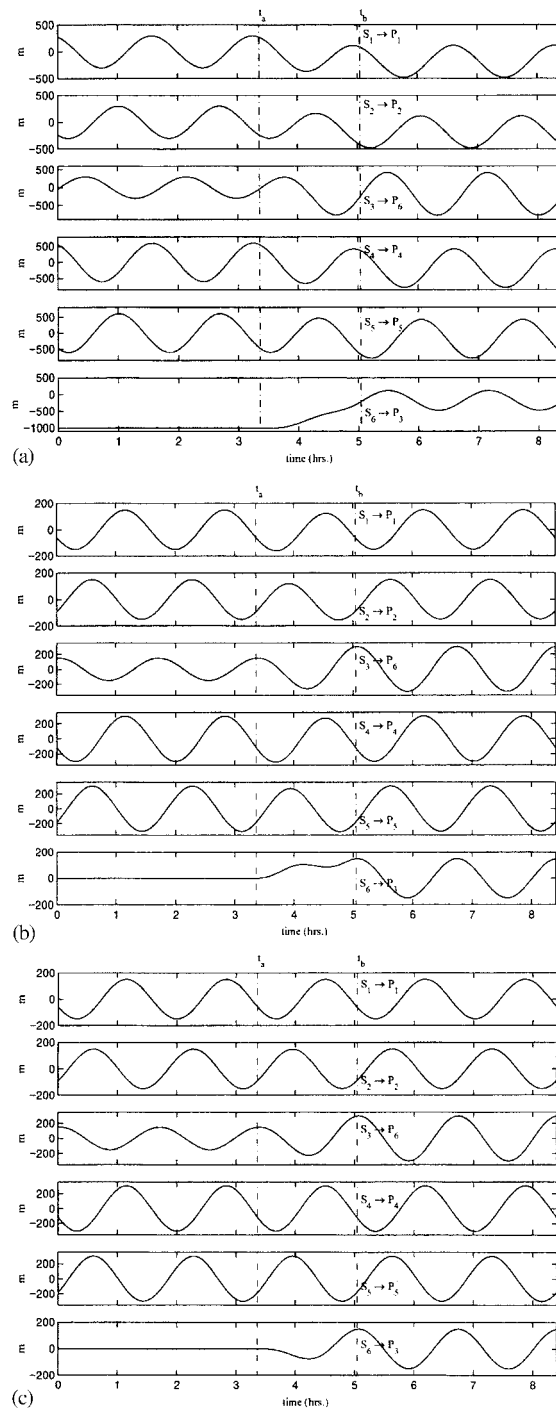
Figure 7. Relative position. (a) x_I, (b) y_I, (c) z_I: Example 1.
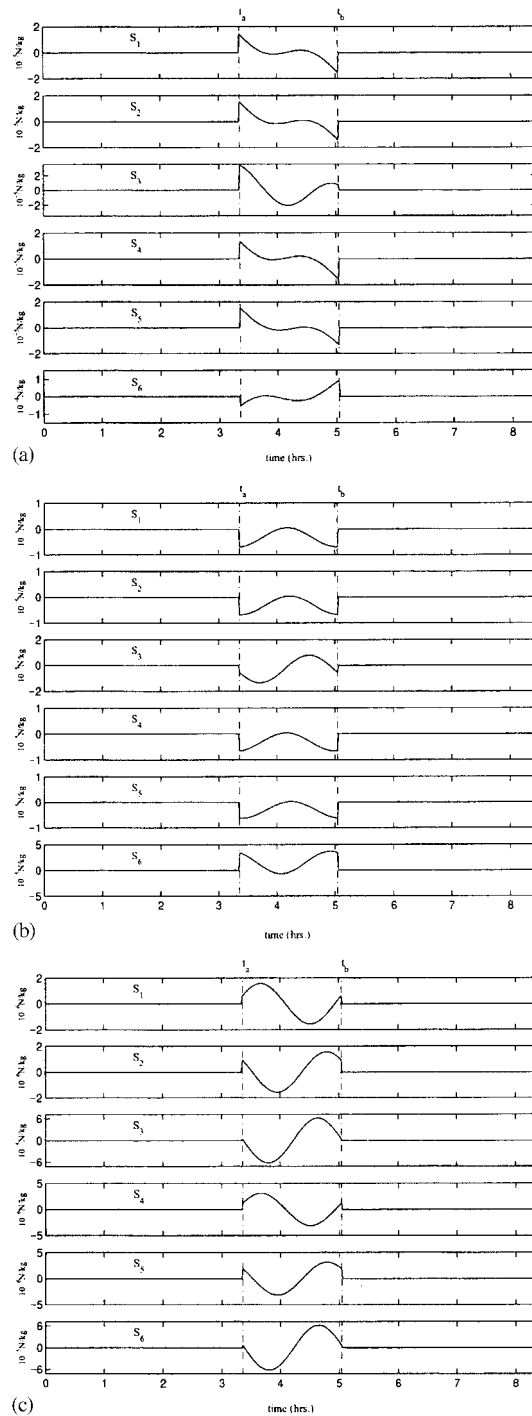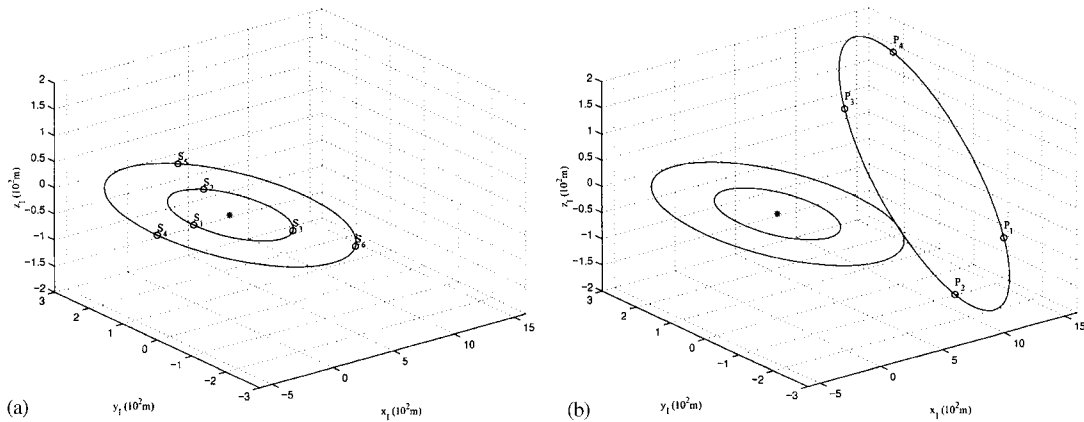
Figure 8. Control input. (a) u_x, (b) u_y, (c) u_z: Example 1.

Table VIII. Spacecraft fuel consumption with optimal permutation: Example 1
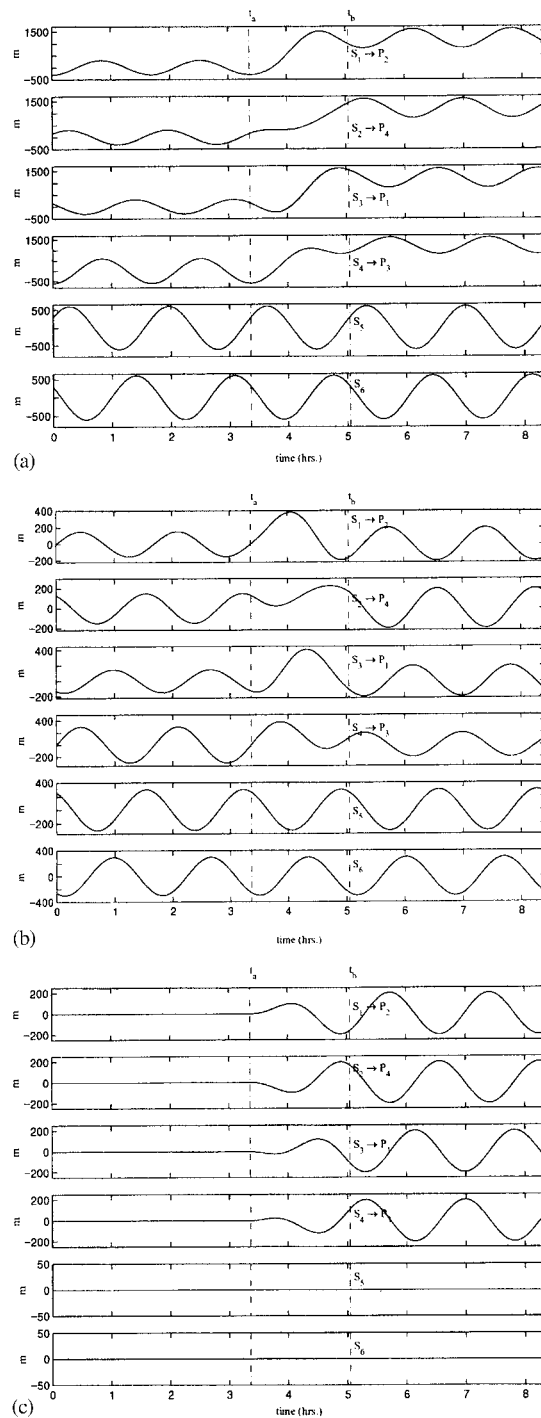
Spacecraft i	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$	Total
Unit: 10^{-3} kg	0.0896	0.0875	2.9238	0.0933	0.0892	4.8546	8.1380

Figure 9. (a) Formation pattern at t_a; (b) desired formation pattern: Example 2 ($\star$ denotes the imaginary leader spacecraft).

Table IX. Formation reconfiguration fuel cost data: Example 2 (unit: 10^{-3} kg).

$I(m_{i,j}^{*}(t; t_a, t_b, \tilde{\lambda}^{*}))$	$j = 1$	$j = 2$	$j = 3$	$j = 4$
$i = 1$	7.1306	8.2036	8.7105	7.1506
$i = 2$	8.2673	14.2557	14.1634	7.6879
$i = 3$	7.0501	12.4297	15.3806	9.5138
$i = 4$	9.0948	7.0934	6.9853	8.4996
$i = 5$	10.8334	18.6632	17.3568	9.0399
$i = 6$	8.3990	15.0112	19.7912	12.6918

Example 5.2

In this example, we assume that initially 6 spacecraft are distributed on two concentric primary formation orbits as shown in Figure 9(a). The initial spacecraft formation consists of two primary formation orbits with the initial thrust free periodic trajectories of spacecraft S_i, $i = 1, \ldots, 6$, given by (72), with $\theta = \pi$. In particular, spacecraft 1, 2, and 3 are distributed on the primary formation orbit specified by (72) with $(X_C, v_{y_0}, v_{z_0}, z_0) = (0, -150\,\omega\,\text{m/s}, 0, 0)$. Furthermore, for spacecraft 1, 2, and 3 we are given $\beta_1 = 0$, $\beta_2 = 2\pi/3$, and $\beta_3 = 4\pi/3$, respectively. Similarly, spacecraft 4, 5, and 6 are distributed on the primary formation orbit specified by (72) with $(X_C, v_{y_0}, v_{z_0}, z_0) = (0, -300\,\omega\,\text{m/s}, 0, 0)$. Furthermore, for spacecraft 4, 5, and 6 we are given $\beta_4 = 0$, $\beta_5 = 2\pi/3$, and $\beta_6 = 4\pi/3$, respectively.

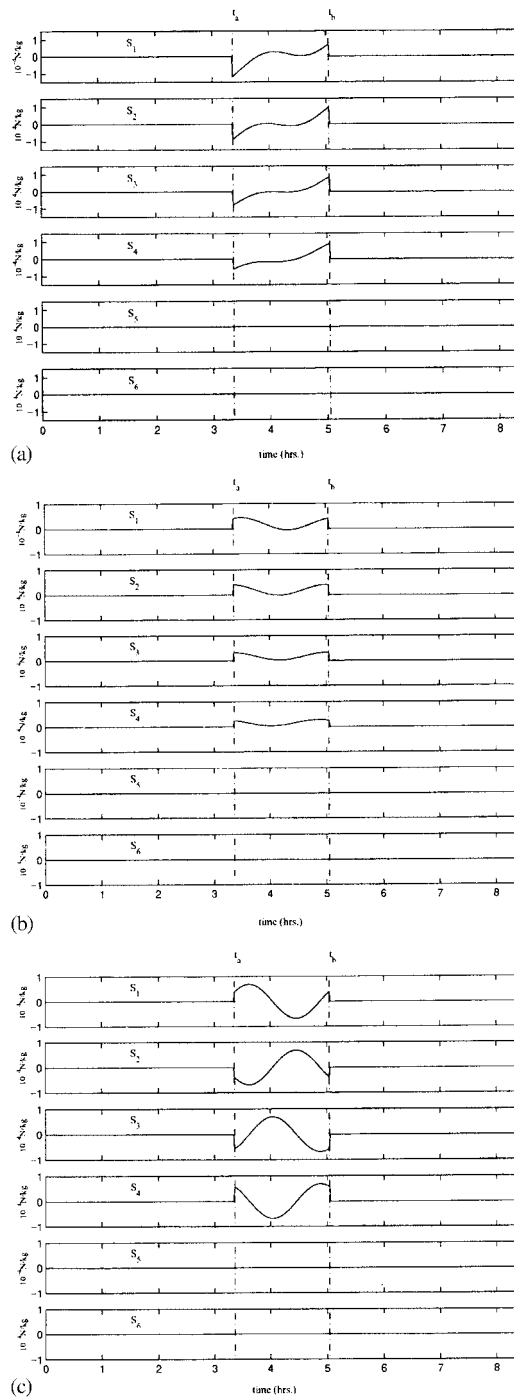Figure 10. Relative position (a) x_I, (b) y_I, (c) z_I: Example 2.

Figure 11. Control input. (a) u_x, (b) u_y, (c) u_z: Example 2.

Table X. Spacecraft fuel consumption with optimal permutation: Example 2.

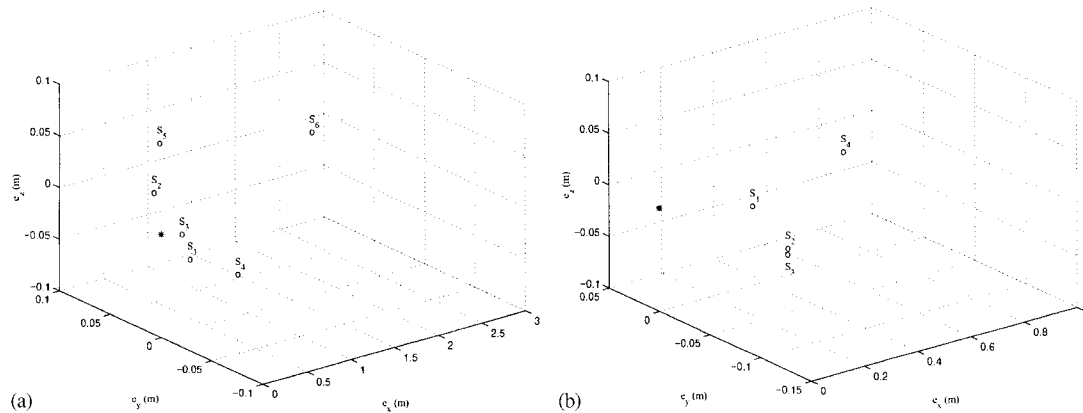| Spacecraft i	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$	Total
Unit: 10^{-3} kg	8.2036	7.6879	7.0501	6.9853	0.0	0.0	29.9269

Figure 12. Target position error under nonlinear simulation: (a) Example 1, (b) Example 2.

Next, 4 spacecraft are required to form a new formation pattern that has four allowable positions distributed on one primary formation orbit (see Figure 9(b)) specified by (73) with $(X_C, v_{y_0}, v_{z_0}, z_0) = (1200 \text{ m}, -200\,\omega \text{ m/s}, -200\,\omega \text{ m/s}, 0)$. Furthermore, for the allowable position 1, 2, 3, and 4 in the new formation pattern $\beta_1 = 0$, $\beta_2 = \pi/2$, $\beta_3 = \pi$, and $\beta_4 = 3\pi/2$, respectively. In this example, θ_P in (73) is allowed to be selected freely from $[0, \pi/2]$ rad.

The optimization algorithm described in Section 4 yields $\theta_P^* = 0.588$ rad for the new formation pattern. The fuel cost for spacecraft S_i, $i = 1, \ldots, 6$, to relocate to the allowable location P_j, $j = 1, \ldots, 4$, in the new formation pattern specified by θ_P^* is given in Table IX. The dynamic-programming-styled task assignment algorithm yields $H^* = [3, 1, 4, 2]$ to be the optimal spacecraft permutation in the new formation pattern specified by θ_P^*. Note that H^* indicates that spacecraft 1, 2, 3, and 4 participate in the new 4 spacecraft formation while spacecraft 5 and 6 continue along their initial thrust free periodic trajectories. Figure 10 gives the fuel optimal spacecraft formation reconfiguration trajectories for spacecraft 1, 2, 3, and 4 as well as the unaltered thrust free trajectories for spacecraft 5 and 6. Furthermore, Figure 11 provides the thrust histories for all the 6 spacecraft. Finally, the total fuel mass consumed by each spacecraft is given in Table X.

Remark 5.1
For the above two examples, utilizing the spherical Earth model leading to nonlinear relative dynamics of spacecraft, we validate our fuel optimal formation reconfiguration algorithm by simulation. Specifically, we apply the optimal thrust program for spacecraft formation

Table XI. Spacecraft fuel consumption accounting for nonlinear dynamics: Example 1.

Spacecraft i	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$	$i=6$	Total
Unit: 10^{-3} kg	0.0897	0.0877	2.9235	0.0937	0.0898	4.8492	8.1336

Table XII. Spacecraft fuel consumption accounting for nonlinear dynamics: Example 2.

Spacecraft i	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$	$i=6$	Total
Unit: 10^{-3} kg	8.1953	7.6732	7.0345	6.9780	0.0	0.0	29.8810

reconfiguration manoeuvres generated by the linear dynamics model (according to Section 4.2) in nonlinear simulation. As shown in Figure 12, under the nonlinear simulations, the offset of each spacecraft at the end of its formation reconfiguration manoeuvre from the desired target position is negligible. Alternatively, we define $\Delta S_i \triangleq \int_{t_a}^{t_b} \Delta v_i(t)\, \mathrm{d}t$, where $\Delta v_i(t) \triangleq \int_{t_a}^{t} |u_i|\, \mathrm{d}t$ for $t_a \leqslant t \leqslant t_b$ and $i \in \{x, y, z\}$. Next, for each spacecraft that participated in formation reconfiguration, we compute $\Delta S = (\Delta S_x^2 + \Delta S_y^2 + \Delta S_z^2)^{1/2}$ and $e = (e_x^2 + e_y^2 + e_z^2)^{1/2}$, where e_x, e_y, and e_z are the x, y, and z components, respectively, of the final position offset under nonlinear simulation. Then, in the two numerical examples considered, for every spacecraft that executed the formation reconfiguration manoeuvre, the value of $e/\Delta S$ is determined to be in the order of 10^{-3} (or smaller). Finally, we modify the optimal thrust program of Section 4.2 to cancel the nonlinear effect of spacecraft relative dynamics. We note that such a modified thrust program eliminates the offset of each spacecraft at the end of its formation reconfiguration manoeuvre from the desired target position without any significant change in the fuel cost (See Tables XI and XII).

6. CONCLUSION

In this paper, we modelled and analysed the distributed spacecraft formation reconfiguration problem as a multi-agent optimization problem. Specifically, we addressed the problem of fuel minimization for spacecraft formation reconfiguration manoeuvres. Based on the qualitative analysis of the formation reconfiguration formulation developed in this paper, we obtained an optimal choice of the manoeuvre time intervals for all the spacecraft. Furthermore, we developed an optimization technique for fuel optimal formation reconfiguration. Specifically, we utilized the theory of calculus of variation, task assignment, and parameter optimization to generate the fuel optimal formation reconfiguration manoeuvres. We obtained the general form of fuel optimal manoeuvre for linearized dynamic model of a spacecraft with VSI propulsion system. In addition, we developed a dynamic-programming-styled task assignment algorithm to assign spacecraft to the allowable positions in the desired formation pattern. A communication protocol has been proposed to implement this algorithm in a distributed manner. Finally, the parameter optimization problem was solved using genetic algorithm. Illustrative numerical simulations were presented to demonstrate the efficacy of the optimization algorithm. Future

work will advance the optimization algorithm to account for perturbation effects, non-circular orbits, and various constraints, e.g. spacecraft permutation constraints, spacecraft distance constraint to ensure collision avoidance, thrust magnitude constraint, etc.

REFERENCES

1. Hedrick JK, Tomizuka M, Varaiya P. Control issues in automated highway systems. *IEEE Control Systems Magazine* 1994; **14**:21–32.
2. Radio Technical Commission for Aeronautics, *Final Report of RTCA Task Force 3: Free Flight Implementation.* Tech. Rep. RTCA, Washington, DC, 1995.
3. Pappas G. *et al.*, A next generation architecture for air traffic management systems. *Proceedings of the IEEE Conference on Decision and Control,* San Diego, CA, 1997, 2405–2410.
4. Kang W, Xi N, Fitch O. Mobile robot tracking control in a perceptive frame. *Proceedings of the SPIE,* Boston, MA, 1998.
5. Scientific Advisory Board, UAV technologies and combat operations, *Tech. Rep.* SAB-TR-96-01, United States Air Force, November 1996.
6. Robertson A, Corazzini T, How JP. Formation sensing and control technologies for a separated spacecraft interferometer. *Proceedings of the American Control Conference,* Philadelphia, PA, 1998; 1574–1579.
7. Tomlin C, Pappas GJ, Sastry S. Conflict resolution for air traffic management: a case study in multi-agent hybrid systems. *IEEE Transactions on Automatic Control* 1998; **43**:509–521.
8. Wang, PKC, Hadaegh FY. Coordination and control of multiple microspacecraft moving in formation. *Journal of Astronautical Science* 1996; **44**(3):315–355.
9. Available on the Web at http://www.vs.afrl.af.mil/factsheets/TechSat21.html and http://www.vs.afrl.af.mil/TechProgs/TechSat21/.
10. *Proceedings of the Air Force Research Laboratory — Formation Flying and Micro-Propulsion Workshop*, Lancaster, CA, 1998.
11. Bauer F *et al.* Satellite formation flying using an innovative autonomous control system (AUTOCON) environment. *Proceedings of the AIAA Guidance, Navigation, and Control Conference,* New Orleans, LA, 1997; 657–666.
12. Guinn JR. Autonomous navigation for the new millennium program earth orbiter 1 Mission. *Proceedings of the AIAA Guidance, Navigation, and Control Conference,* New Orleans, LA, 1997; 612–617.
13. Lau K *et al.* The new millennium formation flying optical interferometer. *Proceedings of the AIAA Guidance, Navigation, and Control Conference,* New Orleans, LA, 1997; 650–656.
14. Leitner J, Beck J, Bell K. Advanced guidance, navigation, and control for remote sensing. 1997, AIAA Paper.
15. Sedwick RJ, Kong EMC, Miller DW. Exploiting orbital dynamics and micropropulsion for aperture synthesis using distributed satellite system: applications to TechSat 21. *Defense and Civil Programs Conference,* Huntsville, AL, 1998, AIAA Paper No. 98–5289.
16. Scientific Advisory Board, New World Vistas Air and Space Power for the 21st Century, 1995, http://ecs.rams.com/afosr/afr/sab/edu/text/any/afrtnwv.htm.
17. Hartman K, Weidow D, Hadaegh F. Management of guidance, navigation, and control technologies for spacecraft formations under the NASA cross-enterprise technology development program (CETDP). *Proceedings of the Flight Mechanics Symposium,* Goddard Space Flight Center, Greenbelt, MD, 1999; 283–294.
18. Chao CC, Pollard JE, Janson SW. Dynamics and control of cluster orbits for distributed space missions. *AAS/AIAA Space Flight Mechanics Meeting,* 1999, Paper No. AAS99–126.
19. Chobotov VA (ed.) *Orbital Mechanics.* AIAA: Washington, DC, 1996; 31–33.
20. Clohessy WH, Wiltshire RS. Terminal guidance system for satellite rendezvous. *Journal of Aerospace Science*, 1960; **27**(9):653–658.

21. de Queiroz MS, Kapila V, Yan Q. Adaptive nonlinear control of multiple space craft formation flying. *Journal of Guidance, Control, and Dynamics* 2000; **23**:385–390.
22. Kapila V, Sparks AG, Buffington JM, Yan Q. Spacecraft formation flying:dynamics and control. *Journal of Guidance, Control, and Dynamics* 2000; **23**:561–564.
23. Redding DC, Adams NJ, Kubiak ET. Linear-quadratic stationkeeping for the STS orbiter. *Journal of Guidance, Control, and Dynamics*. 1989; **12**(2):248–255.
24. Sabol C, Burns R, McLaughlin C. Formation flying design and evolution. *Proceedings of the AAS/AIAA Space Flight Mechanics Meeting,* Paper No. AAS 99–121, 1999.
25. Schaub H, Vadali SR, Junkins JL, Alfriend KT. Spacecraft formation flying control using mean orbit elements. *Proceedings of the AAS Guidance and Control Conference,* Paper No. 99–310, 1999.
26. Vassar RH, Sherwood RB. Formationkeeping for a pair of satellites in a circular orbit. *Journal of Guidance, Control, and Dynamics* 1985; **8**(2):235–242.
27. Yan Q, Kapila V, Sparks AG. Pulse-based periodic control for spacecraft formulation flying. *Proceedings of the American Control Conference,* Chicago, IL, June 2000; 374–378.
28. Hadaegh FY, Lu WM, Wang PC. Adaptive control of formation flying spacecraft for interferometry. *Proceedings of the IFAC Conference on Large Scale Systems,* Rio Patras, Greece, 1998; 97–102.
29. Yan Q, Yang G, Kapila V, de Queiroz MS. Nonlinear dynamics, trajectory generation, and adaptive control of multiple spacecraft in periodic relative orbits. *Proceedings of the AAS Guidance and Control Conference,* Paper No. 00–013, Breckenridge, CO, 2000.
30. de Queiroz MS, Yan Q, Yang G, Kapila V. Global output feedback tracking control of spacecraft formation flying with parametric uncertainty. *Proceedings of the IEEE Conference on Decision and Control*, Phoenix, AZ, December 1999; 584–589.
31. Morton B, Weininger N, Tierno JE. Collective management of satellite clusters. *Proceedings of the AIAA Guidance, Navigation, and Control Conference,* Portland, OR, August AIAA Paper No. 99–4267, 1999; 1576–1584.
32. Yan Q, Yang G, Kapila V, de Queiroz MS. Nonlinear dynamics and output feedback control of multiple spacecraft in elliptical orbits. *Proceedings of the American Control Conference,* Chicago, IL, June 2000; 839–843.
33. Yeh H-H, Sparks, AG. Geometry and control of satellite formations. *Proceedings of the American Control Conference,* Chicago, IL, June 2000; 384–388.
34. Kreyszig EW. *Advanced Engineering Mathematics.* Wiley: New York, NY, 1998.
35. Hill PG, Peterson CR. *Mechanics and Thermodynamics of Propulsion.* Addison-Wesley: Reading, MA, 1992.
36. Luenberger DG. *Optimization by Vector Space Methods.* Wiley: New York, NY, 1969.
37. Kirk DE. *Optimal Control Theory.* Prentice-Hall: Englewood Cliffs, NJ, 1970.
38. Pierre DA. *Optimization Theory with Applications.* Dover: New York, NY, 1986.
39. Bellman R. *Dynamic Programming.* Princeton University Press: Princeton, NJ, 1957.
40. Belegundu AD, Chandrapatla TR. *Optimization Concepts and Application in Engineering.* Prentice-Hall: Upper Saddle River, NJ, 1999.
41. Hillier FS, Lieberman GJ. *Operations Research.* Holden-Day Inc.: San Francisco, CA, 1974.
42. Davis L. *Handbook of Genetic Algorithms.* Van Nostrand Reinhold: New York, NY, 1991.
43. Goldberg DE. *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley: New York, NY, 1989.
44. Holland JH. *Adaptation in Natural and Artificial Systems.* University of Michigan Press: Ann Arbor, MI, 1975.
45. Krishnakumar K. Micro-genetic algorithms for stationary and non-stationary function optimization. *SPIE: Intelligent Control and Adaptive Systems,* Philadelphia, PA, vol. **1196**, 1989; 289–296.
46. Zalzala AMS, Fleming PJ. *Genetic Algorithms in Engineering Systems.* IEE Control Engineering Series, Vol. 55, 1997.
47. Srinivas M, Patnaik LM. Adaptive probabilities of crossover and mutation in genetic algorithm. *IEEE Transactions on Systems, Man and Cybernetics* 1994; **24**(4):656–667.